Pers Ubiquit Comput (2018) 22:333–344
DOI 10.1007/s00779-017-1067-4

ORIGINAL ARTICLE

Data provenance to audit compliance with privacy policy in the Internet of Things

Thomas Pasquier[1] · Jatinder Singh[2] · Julia Powles[3] · David Eyers[4] · Margo Seltzer[1] · Jean Bacon[2]

Received: 3 March 2017 / Accepted: 12 June 2017 / Published online: 15 August 2017

Abstract Managing privacy in the IoT presents a significant challenge. We make the case that information obtained by auditing the flows of data can assist in demonstrating that the systems handling personal data satisfy regulatory and user requirements. Thus, components handling personal data should be audited to demonstrate that their actions comply with all such policies and requirements. A valuable side-effect of this approach is that such an auditing process will highlight areas where technical enforcement has been incompletely or incorrectly specified. There is a clear role for technical assistance in aligning privacy policy enforcement mechanisms with data protection regulations. The first step necessary in producing technology to accomplish this alignment is to gather evidence of data flows. We describe our work producing, representing and querying audit data and discuss outstanding challenges.

1 Introduction

The Internet of Things (IoT) is projected to be a multi-trillion dollar industry with considerable potential to revolutionise a wide range of sectors, including health, cities, factories and energy [41]. Realising the broad IoT vision entails data sharing, often in a user-centric and ad hoc manner, across a range of technologies, platforms and providers [27]. At present, we see that IoT applications tend to operate within silos, as defined by manufacturers, service providers and/or the associated technological stack. Realising the broader IoT vision makes interoperability and establishment of standards a requirement [36]. Here, we focus on societal issues, in particular, the privacy of personal data generated during IoT processes.

IoT devices and their enabling systems are, by their nature, a constant witness to our everyday lives, being deployed throughout public and private spaces. We are already trackable electronically, e.g. through using credit cards and mobile phones, but there are several mechanisms for exercising a degree of control. In IoT, tracking is universal and ubiquitous, which threatens to mark the dawn of a new era, where every detail of one's life is monitored, captured and analysed, potentially in real-time. The availability of increasingly sophisticated technology for image processing, learning and inference, means that people will be

The software artefacts corresponding to our work to date are available online under a GNU General Public Licence at http://camflow.org/.

✉ Thomas Pasquier
tfjmp@seas.harvard.edu

Jatinder Singh
Jatinder.Singh@cl.cam.ac.uk

Julia Powles
julia.powles@cornell.edu

David Eyers
dme@cs.otago.ac.nz

Margo Seltzer
margo@eecs.harvard.edu

Jean Bacon
Jean.Bacon@cl.cam.ac.uk

[1] Center for Research on Computation and Society, Harvard University, Cambridge, MA, USA

[2] Computer Laboratory, University of Cambridge, Cambridge, UK

[3] Computing and Information Science, Cornell Tech, New York, NY, USA

[4] Department of Computer Science, University of Otago, Dunedin, New Zealand

identified from the gathered data and their movements and personal interactions monitored. Such data is personal and private, and in most countries subject to law and regulation.

Many aspects of the IoT are consumer driven. For the IoT to succeed, people and organisations must accept and be prepared to pay for IoT technology, whether through their money or their data. They must therefore have confidence in the performance of connected devices and systems (including security), trust in the protection of private information, and realistic, traceable options for opt-out. Such concerns are not only consumer driven; data protection and privacy laws in many parts of the world mandate user consent and control over personal data. To satisfy these requirements, it is essential that systems and devices not only perform appropriately in a secure manner, but that there is transparency and accountablity, i.e. that it is possible to observe system behaviour (transparency) to verify in a tangible, accessible manner that user and system preferences have been met in accordance with regulatory requirements (accountability). Moving forward, there must be transparency in the form of evidence for users to understand how, when and why their personal data, or others' data for which they are responsible, is being used, and in what contexts. To achieve this, many levels of work are required, including how to specify policy that embodies or reflects law and regulation, and how to design user interfaces to ascertain their preferences. We focus on how data flow and usage may be audited, so that compliance with law and regulation can be verified.

Currently, there are few technical mechanisms for enabling this. Current practice for the use of web services is often that the "small print" of the terms and conditions of use explicitly ask users to effectively agree to waive their various rights to privacy and data protection. User metadata and sometimes even content may be used for commercial purposes, such as through data analytics—"if the service is free, you are not the customer, you are the product", as the adage goes. For the cyberphysical world of the IoT, such practices need to be considered in the light of existing and emerging regulation, as Section 3 discusses.

We consider some of the concerns that regulators in Europe and the US have already raised about IoT functionality. We generalise these concerns to a primary technical challenge: to ensure swift, accountable realisation of appropriate data flows. These principles that we seek to enforce derive from both data protection and privacy requirements, and are designed to accomodate the wishes and expectations of users, system managers, and third parties. Requirements for data protection will often differ depending on the environment, e.g. home, workplace, hospital and a variety of public spaces. These requirements can be expressed as policies regarding the flow of data, which must be enforced and shown to have been enforced [59]. Here we focus on means to record information flow from runtime execution that can be used as an audit trail to demonstrate compliance with specified policy. Such audit can be represented as data provenance [14], a model that represents interaction between data items, processes and individuals as a directed acyclic graph. Data provenance can be analysed to investigate suspected security breaches and monitor compliance with security policy [9, 10], or to verify run-time properties of a system [34].

We explore the challenges of auditing data flows throughout the IoT. Best practice in future IoT will require such evidence as a basis for transparency and accountability. Data-flow audit is not a panacea but an essential first step. Work is also required on policy specification and enforcement, interfaces for ascertaining users' privacy preferences, and interfaces for various parties to investigate the audit. We address how audit data can be gathered, how audit in a large-scale system such as the IoT might be managed and how graph processing tools might assist in querying the audit data.

2 Internet of Things

The IoT, though currently the subject of much hype and promise, is not a term with a formal definition. The ITU

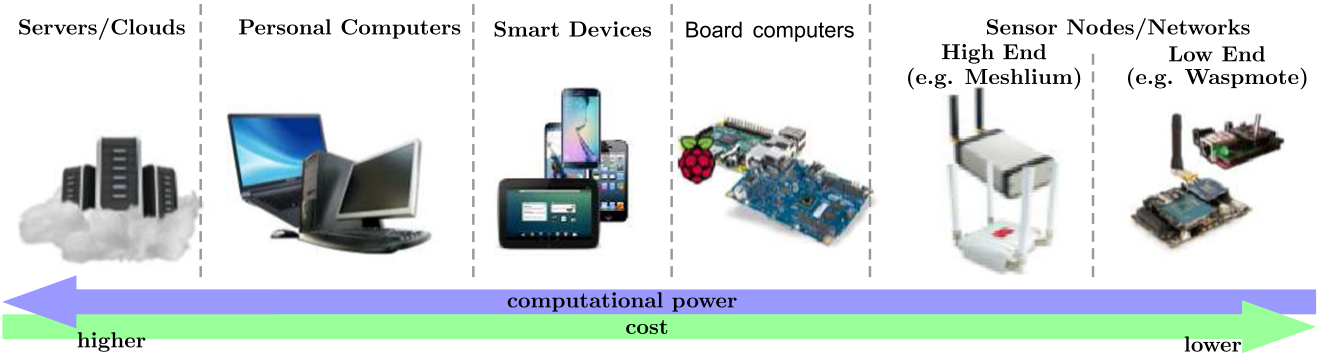

Fig. 1 Informal IoT devices categorisation by price and computational power

[1] describes the IoT as: *A global infrastructure for the information society, enabling advanced services by interconnecting (physical and virtual) things based on existing and evolving interoperable information and communication technologies.* From this, we extract an important concept: the interconnection of *physical or virtual* things in an *interoperable* fashion. IoT applications have the potential to integrate a large spectrum of devices, of various resource capabilities (Fig. 1), often provisioned through cloud services [42]. These aspects raise important considerations for any management technology. Although there has been significant work on wireless sensor networks, as well as infrastructure for supporting ubiquitous systems in general; the wide-scale interoperability, in line with the broader vision of the IoT, has yet to be realised [58].

For reasons ranging from customer lock-in, to legal, technical and security concerns, many currently deployed (so-called) IoT systems could be better described as *silos of things*. That is, services tend to be of a closed nature, where interactions between services are often limited to a number of known types of things and a number of known services. These silos can be undesirable customer lock-ins due to non-standard technology and software. More generally, management domains or application contexts may structure the IoT in a desirable way through being subject to a domain/context-wide authority for policy definition and enforcement. Examples are a smart home or a public space.

To achieve the broader IoT vision, it is necessary to consider how the flows of data within and outside such contexts can be negotiated and controlled. To this end, we have investigated the use of information flow control [52] for system-wide control of data flows and made the case that such an approach is relevant for the IoT [59]. Here, we focus on two things: (1) audit and data provenance to achieve transparency on where data has flowed and (2) the analysis of audit data to demonstrate regulatory compliance (facilitating accountability).

We consider a future in which technology and standards exist for the composition and interoperation of things, thus realising the broader IoT vision from a technical viewpoint. But for a legally compliant IoT, it is also necessary to address the fact that much of the data gathered by IoT becomes personal as soon as people are identifiable, and therefore becomes subject to data protection law [29]. To comply with this, policy must be defined and enforced on how data flow can and should be controlled during such interactions, and compliance must be demonstrated through audit. The vast scale of the IoT makes audit a major challenge. We show how the structure of the IoT into application contexts and management domains can be captured to make audit feasible.

3 Legal context

As introduced in Section 1, the gathering and interpretation of personal data from IoT devices raises significant privacy concerns [29, 64]. These concerns compound the challenges introduced by cloud computing [62] and big data analysis [60], and have made the IoT a key priority for privacy and data protection regulators [37]. The IoT is also of great interest to competition authorities and consumer protection and safety bodies, but here we restrict our analysis to the privacy dimension of law.

Data protection laws, led by Europe and adopted by many countries around the world, seek to regulate and control all flows of personal data (in essence, information identifiable to individuals) to specific legitimate purposes, with various safeguards for individuals and responsibilities on those who hold, manage and operate on personal data. In other jurisdictions, most notably the USA, which do not have such omnibus data protection laws, there are sectoral restrictions on personal data in areas such as health and finance, as well as general Fair Information Practice Principles (FIPP), which include principles such as notice, choice, access, accuracy, data minimization, security and accountability.

In recent years, regulators on each side of the Atlantic have paid attention to the IoT as an emerging phenomenon and challenge to privacy and data protection. We draw particularly on two reports made by the leading regulatory authorities in Europe and the US, grappling with the IoT as a direct subject of interest: (1) an Opinion issued in September 2014 by European regulators under the umbrella of the Article 29 Data Protection Working Party (WP29); and (2) a Staff Report in January 2015 (and reiterated in comments in June 2016 to a Department of Commerce Request for Comment) by the US Federal Trade Commission (FTC).[1]

Both the WP29 and FTC reports emphasise the continued applicability of existing laws to the IoT. Of particular interest for our purposes, they also cohere on two main points that we extend throughout our analysis. The first is a recognition that *changes in the context of personal data flows* demand user involvement, i.e. between different environments, with different parties involved, or towards different ends. Notably, this does not apply if data is de-identified immediately and effectively; though if this course is taken and data is re-identified, responsibility must follow. Audit of data flow assists in demonstrating how data is used after its release. The second point made in both reports is that user

[1] https://www.ftc.gov/system/files/documents/reports/federal-trade-commission-staff-report-november-2013-workshop-entitled-internet-things-privacy/150127iotrpt.pdf, https://www.ftc.gov/system/files/documents/advocacy_documents/comment-staff-bureau-consumer-protection-office-policy-planning-national-telecommunications/160603ntiacomment.pdf

involvement may be difficult in an IoT ecosystem, given the scale of data flows, the diversity of potential interactions, and the frequent absence of a consumer interface, but that this does not diminish the responsibility to provide *effective notice and control* to users ("clear, prominent and conspicuous", according to the FTC; "clear, comprehensive and user-friendly" according to the WP29) on flows of personal data throughout the IoT.

As an example, the FTC elaborates on the data that an application should gather from a wearable device as follows: As an example of how data minimization might work in practice, suppose a wearable device, such as a patch, can assess a consumer's skin condition. The device does not need to collect precise geolocation information in order to work; however, the device manufacturer believes that such information might be useful for a future product feature that would enable users to find treatment options in their area. As part of a data minimization exercise, the company should consider whether it should wait to collect geolocation data until after it begins to offer the new product feature, at which time it could disclose the new collection and seek consent. The company should also consider whether it could offer the same feature while collecting less information, such as by collecting zip code rather than precise geolocation. If the company does decide, it needs the precise geolocation information, it should provide a prominent disclosure about its collection and use of this information, and obtain consumers' affirmative express consent. Finally, it should establish reasonable retention limits for the data that it does collect.

The context surrounding data is also an important consideration. Context has been considered pre-IoT regarding access to Electronic Health Records (EHRs) and personal fitness monitoring [5, 6, 28]. A person may have specified a background access control policy, defining who (what role) can access their medical data. A change in context arises when someone has a medical emergency while exercising outside the home, or due to a traffic accident. Ideally, their policy should indicate what access can be made to their data, even when they are unconscious in an emergency situation. If this is not the case, an emergency override may be made, sometimes known as a "break-glass policy". Here, audit is seen as essential, as a safeguard that the decisions taken were in the best interests of the patient (data subject). A similar example from the IoT is that an internet-connected fitness monitor may indicate that its owner is suffering a medical emergency, causing an application context change for use of the data it is gathering, from lifestyle to medical.

One of the ways that data protection laws deal with the complexity of IoT services is to impose stringent responsibilities on those who determine the purpose and manner of data collection and use (known as *data controllers*), as well as those who hold, manage, and operate on personal data on their behalf (*data processors*). This ensures the reach-through of responsibility for proper data handling, but it can be onerous when a data controller has little control or view of the data processor's internal workings [21]. In Section 4, we argue that the scope of responsibility of data controllers and processors might best be defined by structuring the IoT, where appropriate, as a federation of administrative domains.

We explore the contribution that audit tools can make in addressing both points above: (1) that changes in the context of personal data flows demand user involvement; (2) that although involving users is difficult, this does not diminish the responsibility to provide effective notice and control. Audit tools may also contribute to meeting a range of data protection law obligations, in particular:

- **Transparency:** informing users about the identity of the data controller, the purposes of the processing, the recipients of the data (including use for direct marketing purposes and possible sharing with specified categories of third parties), use of sensitive data, and the existence of users' rights of access, opposition and discontinuation of service. Audit tools can indicate to users to where their data has flowed, how it has been used and processed, and who has accessed it.
- **Security:** implementing and sufficiently guaranteeing appropriate technical and organisational measures to protect personal data, as well as performing security assessments of systems as a whole, applying principles of composable security. Note that these obligations tend not to prescribe the use of any particular (technical) security techniques. Audit tools can assist in demonstrating that security measures have been taken appropriately.
- **Enabling users' rights:** facilitating the user's rights of access to raw data and intelligible information about how their data is processed and any decisions made from it; rights to opposition; rights to modification and deletion of personal data, at a fine-grained level for each type of data collected by a specific thing, the same type of data collected by different things, or a particular operation on all personal data; and rights to discontinue a given service. Note that audit of data flows is necessary to ascertain whether these rights have been adhered to through policy expression and enforcement, thus helping to demonstrate compliance.

3.1 Examples: data flows within and between application contexts

If application contexts were to be opened up, a person's movements from home, when travelling by car or public transport, at work, at lunch, in the park, at the cinema etc. could potentially be publicly available. This would likely be an invasion of privacy and require regulation. Similar issues already arise regarding mobile phone tracking, when a person's location might be available to limited numbers of people, such as a controlling family member as well as the phone company.

Within some application contexts, such as the home and workplace, identities may acceptably be recorded for internal use. Within a cinema or restaurant, public identification of staff and customers will generally be unacceptable. Controlled identification of, say, staff to management but not of customers may be needed. In a smart city with traffic control, it may be that only the police and not the general public may know the identities of drivers, but even this is fraught with difficulty in the context of racial profiling in America. More generally, the presented examples are illustrative and the regulation concerns are far more nuanced.

Regarding regulation of the opening up of application contexts, we need to consider what data is legitimately needed to flow from them. When a train arrives at a station, coarse grained information such as the number of people exiting the station is useful for scheduling taxis and buses, while information identifying individuals is superfluous and can be considered excessively invasive. On the other hand, in the case of an emergency in a public building such as a fire alarm or bomb scare, it is desirable to be able to identify individuals.

An audit mechanism must be able to identify (1) the state of data items, including data resulting from transformations (e.g. an aggregate vs. identifying information); (2) the service accessing the information; and (3) the context in which the data is being accessed. Such information can be derived from the analysis of data-provenance graphs, as discussed in Section 6.

4 The challenges of system-wide policy

Existing technical mechanisms, such as access control, have been proposed to control the use and flow of data beyond an individual's direct control [70] or to maintain certain bounds around data, through managing storage and computation specifics [15, 61].

Proposed mechanisms for technically enforcing security policy in the IoT include authentication, remote attestation, access control and encryption; see [58]. Such mechanisms may be used to comply with specific legal requirements, and some specific technical approaches can be required by law, for instance, that medical data must be stored in an unintelligible (encrypted) form.

Generally, there are a number of issues concerning mechanisms for policy enforcement for IoT [59]:

- A uniform enforcement mechanism is unlikely to be possible across solutions or administrative domains, even in situations where service providers and device manufacturers make best efforts. For example, a front-end service may offer individual users modifiable privacy settings, while the underlying storage service may only be able to provide access guarantees per application, and not for individual users of the application.
- The context in which a device is used may define the regulatory constraints to which it is subject. The ad hoc and user-driven nature of the IoT vision may allow a device originally designed for a particular purpose (e.g. lifestyle monitoring) to be used in another domain (e.g. medical), subject to a different set of concepts, constraints and regulatory frameworks. Even similar products may express and enforce privacy settings in different ways (a familiar example is the inconsistency of privacy settings across different social media).
- There is generally very little means to control, monitor or audit the use of data after it has been allowed to leave some application context. This is especially true for end users who may not fully understand or be aware of the complex chain of providers involved in the delivery of a service. However, the law has the concept of "reach-through" and the requirement that it is enforced. Audit is a first step towards complying with this requirement.

Such issues may naturally lead to discrepancies between regulatory requirements, including user preferences, and the tools deployed technically to enforce those requirements and preferences, as illustrated in the planes of Fig. 2. In deployment, mismatches between the regulation, enforcement and audit planes in Fig. 2 are inevitable. This is because the enforcement mechanisms may not perfectly align with the regulation plane, due to the restricted scope of technological solutions, differences in interpretation of the law, system-specific requirements and constraints, end-user or economic pressures etc.

Therefore, service providers may decide to limit interactions and exchange of data outside a number of well-defined

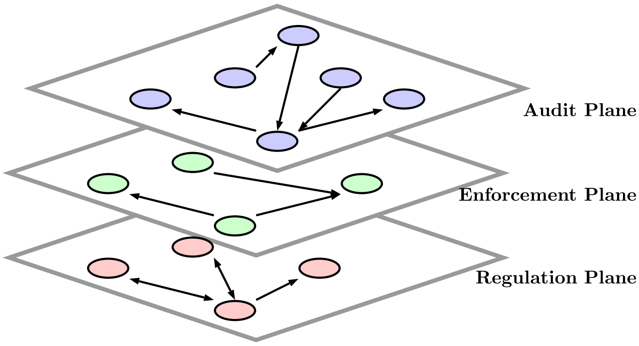

Fig. 2 Data flow planes

services, in order to meet their security obligations (see Section 3) among other considerations (see Section 2). This can be regarded as creating application or administrative domains, preventing data from flowing more widely without negotiation. We argued in Section 2 that the silos that lock users in to specific technologies are undesirable for meeting the wider IoT vision. Given interoperability standards, structuring the IoT as federated administrative domains is a likely natural consequence to facilitate management. In Section 5 we see that representing such a structure in an audit graph is a means for managing the scale of audit in the IoT.

Further, as discussed in Section 3, even a perfect enforcement mechanism that fully implements defined policy on allowed data flows, would not of itself meet the emerging requirements for transparency and accountability. To achieve these, an audit mechanism is also required to provide supporting evidence: (1) as a basis for the mandated transparency for end users to exercise their rights, (2) to allow trust establishment across administrative domains through mutual auditing, (3) to allow the alignment over time of technical enforcement mechanisms with regulation and end users' requirements.

Traditional application-centric auditing would suffer similar issues to those discussed above for the enforcement mechanism: focusing only on the aspect important for a particular application and hindering understanding of system-wide behaviour. These issues make meaningful exploitation of such audit data difficult system-wide. It appears that in order to align with the information-centric nature of the regulation and end-user privacy requirements, an information-centric audit mechanism needs to be exploited. This is the third (audit) plane in Fig. 2, representing information exchange resulting from actual executions. In Section 5, we discuss technical means and challenges in order to capture audit data system-wide, and in Section 6, we discuss technical means and challenges when interpreting the audit data.

5 Capture and exploitation of provenance data

Previous sections described how some aspects of data protection law can be expressed as constraints on information flow. This section discusses data provenance as a mechanism to enable transparency over information flow, while Section 6 explores provenance as a means to enable the audit of those information flows.

Provenance [14] is a record of the origin of and transformations applied to data within a system. Provenance aims to answer the following questions: Where do data come from? Who manipulated the data? What transformations were applied? Provenance data can be represented as a directed acyclic graph describing the relationships among elements composing a system (data items, processing steps, users, contextual information etc.). These elements fall into three categories: entities (i.e. data items), activities (i.e. transformations applied to data) and agents (i.e. persons in the legal sense).

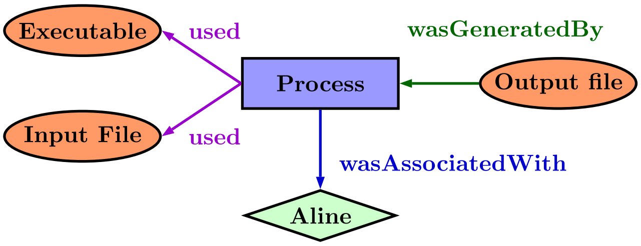

Fig. 3 A simple W3C ProvDM compliant provenance graph

Figure 3 shows a simple provenance graph following the W3C-specified standard [43].[2] This provenance graph represents a process that used an executable and an input file to generate an output file. This process was associated with the user Aline. A provenance graph captured at the OS level by systems such as LPM [10] or CamFlow [49, 51] can contain millions of nodes.

Provenance can be divided into the two broad categories of observed and disclosed provenance [11].

- Observed provenance is captured at the system level, and recently led to "whole-system provenance" [10, 51, 53] that captures all interactions between processes and the operating systems, aiming for completeness. In a Linux context, completeness can be ensured through guarantees provided by Linux Security Modules [20, 23, 32]. We rely on such properties in our own implementation (CamFlow) [51]. CamFlow can be used in systems running some Linux distributions (from servers, to low-end devices) or to smartphones/tablets running the Android OS. In closed systems (e.g. Windows or MacOS), weaker coverage has been achieved (i.e. fewer information flow sources may be recorded) in projects such as Spade [24].
- Disclosed provenance is provided by applications (as opposed to being generated by the underlying systems) in order to describe inner data dependencies. Disclosed provenance has for example, been proposed for Hadoop MapReduce [2, 18] and Spark [31]. Completeness or correctness of disclosed provenance is harder to guarantee [40]. However, when compared with observed provenance, it is possible to describe semantic information more finely.

[2] https://www.w3.org/TR/prov-overview/, as stated in Section 2 building on standards is of fundamental importance for the interoperability of IoT systems.

Whole-system provenance solutions tend to provide a means to integrate disclosed provenance with system-observed provenance to refine the end results [10, 51]. The Core Provenance Library [40] aimed at integration of provenance from different layers and sources, allowing provenance objects to be referenced and queried in a uniform manner.

Our proposal is based on the following idea: regulation and end users' requirements represent *expected system behaviour*; technical enforcement mechanisms represent *permissible actions*; while provenance data represents *actual system behaviour* [8, 52]. Through analysing the provenance, we can determine if the intent of the regulation is being captured by the enforcement mechanism. Discrepancies between the observed behaviour and expected behaviour can be reported, and the enforcement mechanism corrected accordingly. In the rest of this section, we discuss the challenges posed by wide scale provenance capture.

5.1 Confidentiality: controlling access to provenance data

Provenance records in themselves may constitute sensitive information. Indeed, records such as a history of system execution contain information about a user's activities, and how she interacted with other users. One can learn from ISP records which websites Aminata visited, but not the content of the exchanges. Similarly, provenance records can show that there were interactions between Aminata's and Bernardo's smart watches, even if the content of the messages exchanged is unknown. Access to provenance information must therefore be controlled.

In the literature, this is known as Provenance Access Control [13, 47] (PAC), not to be confused with Provenance-based Access Control [9, 48] (PBAC).[3] PAC must (1) be fine grained; (2) consider privacy constraints; (3) ensure that useful information can be obtained, even when full access cannot be granted. Concerning the last point, a provenance graph with a “hole” (omission), due to access control, might appear to defeat the use of provenance as an audit tool. A solution often proposed in the literature is the abstraction of a provenance graph [30, 44] that both hides sensitive information and conserves the semantic information necessary for provenance analyses.

A more general problem in the IoT is to devise a decentralised access control model [28] that can be adapted to the specifics of provenance data. To the authors' knowledge, this is a challenge that remains to be addressed for the IoT. Our work on access control for widely distributed systems [5] suggests a structure of federated administrative domains with negotiated inter-domain access.

5.2 Integrity: trusting the provenance data

If data provenance is to be used as a primary source of information for audit of the complex behaviour of IoT systems, it is necessary to establish trust in the data. The work of Bates et al. [10] represents the state of the art in the domain. They combine remote attestation [17] based on hardware roots of trust through the Linux Integrity Measurement Architecture [33, 56]; and cryptographic techniques to guarantee non-repudiation of provenance data. Remote attestation is necessary to establish the trustworthiness of the provenance data. Non-repudiation is important since provenance *“records history, and history does not change”* [12]; that is, provenance should include some immutability guarantees. A standard technique would be to use hashing and signing, based on a Public Key Infrastructure (PKI).

5.3 Availability and scalability

Availability is a challenge to be addressed when capturing provenance at the IoT scale; i.e., one must be able to access the information necessary to perform an audit at any given time. Availability issues have generally not been considered when building provenance systems [57]. However, the push towards high-performance provenance, with projects building on top of Apache Accumulo [45],[4] may represent a first step in this direction. Indeed, cloud-based storage systems such as Accumulo are designed with availability as one of their core requirements. Guaranteeing both availability and secured access to provenance across application administrative domains remains an open challenge.

Provenance systems (as well as audit/logging systems in general [22, 35]) generate a very large amount of audit data [14]. Mechanisms must exist to handle the high volume of data generated, so that such systems can sustain a constant ingest of large amounts of data, and not collapse under the workload.

One approach is to use stream processing of the provenance data (e.g. [16]). That is, queries could be applied to provenance data as they are generated. Our own provenance capture in CamFlow allows for the collection of provenance across a distributed system via the publication of provenance data over messaging middleware such as

[3] PBAC uses provenance information to make primary data access decisions, while PAC controls access to the provenance data itself.

[4] Apache Accumulo is a scalable open-source key/value store implementation based on the design of Google's BigTable.

Apache ActiveMQ,[5] MQTT[6] or Apache Flume.[7] Selecting the appropriate messaging middleware is dependent on where in the IoT spectrum the application lies (see Fig. 1). In complex multi-application scenarios it is likely that multiple protocols will need to be supported. Hardware constraints on devices interacting with the physical world, and network accessibility, such as devices' interactions with firewalls and NAT, also need careful consideration when selecting a protocol.

Such a provenance stream can be exploited at scale using tools such as Apache Spark [4] at runtime. For example, an auditor may want to specify a number of queries relating to some regulations (we further discuss the exploitation of provenance data in an audit context in Section 6). Stream processing of the provenance data can lead to the generation of an event that triggers an action (e.g. to alert a customer that a new service has been given access to its data). In such cases, it may not be necessary to retain the entirety of the provenance graph, but only a smaller subset relating to an event.

Provenance can also be stored and exploited at rest. The ingest of a large amount of provenance data is the subject of active research [45]. Provenance being a directed graph, advances in graph processing techniques, e.g. [25, 26, 38, 55, 69] can be leveraged for its analysis.

Means to reduce the amount of collected data have been explored. Bates et al. [8] introduced the *"take only what you need"* approach to provenance where MAC policies are used to determine sensitive objects. Our own work built on Information Flow Control policy [51] to achieve a similar objective. Security policies are used as a filtering mechanism on the audit data. The justification for this is that certain (unlabelled) data are not considered sensitive and need not be the subject of audit. In later work [49, 51], we expanded this concept to multiple dimensions beyond security policies, collecting provenance based on, for example, network interfaces or control groups. The trade-offs of such an approach need to be considered with care. While significantly reducing the amount of data provenance generated, important information may not be recorded. The purpose of the provenance capture needs to be understood and clearly defined, the potential adversary identified, and thus the *minimum* and *sufficient* information required to demonstrate compliance. It remains to be explored in such a complex ecosystem as the IoT, if data items of interest can reliably be identified. For example, inferences over diverse types of public data may reveal sensitive information about people, which argues for all flows, labelled and unlabelled, being audited.

Another approach to deal with the large amount of data is compression. The W3C-PROV standard is W3C RDF-compliant, greatly helping with the processing of the generated graph [7, 46]. Graphs, especially RDF graphs, can be compressed through automatic pattern recognition. Repeating patterns in a graph are identified at write time. Instead of storing multiple representations of each node and edge composing the pattern, the identifier corresponding to the pattern and a list of parameters are stored. Applying the parameters to the pattern allows the subgraph to be recovered at read-time. Techniques specifically tailored to provenance graphs have been explored, leading to a significantly smaller and queryable storage format [67, 68].

5.4 Distributed management

For the entire IoT, whole-system provenance is infeasible and structuring of captured provenance data in line with the various management structures and application contexts within IoT is natural and desirable. The model of application contexts creates a useful partitioning of audit data. Audit data from within the context can be held and investigated separately from external flows between contexts. The global audit graph can represent an application context, such as a workplace, as a single node, capable of expansion if needed. Such a partitioning can also be the basis of access control to the audit data.

In previous work on role-based access control (RBAC), within a structure of federated administrative domains [5], we proposed that inter-domain access should be negotiated in terms of roles defined within the interoperating domains. For example, a doctor at a hospital may be allocated the privileges associated with a role "research-scientist" at a Research Institute for her specialism. For the IoT, given standards for interoperability, we believe that such a structure has potential as a basis for managing audit.

While data provenance has been the subject of extensive research, these results have yet to be combined in a system able to handle provenance at the IoT scale. These issues represent important and interesting research challenges yet to be fully addressed.

6 Verifying compliance with policy

Retrospective security [3, 39, 54, 65] is the detection after execution of security violations. That is, suspicious transactions are not actively prevented. Retrospective security can be motivated by the difficulty of enforcing policy consistently across a complex system as discussed in Section 4 and involvement of elements outside the system (third

[5] http://activemq.apache.org/

[6] ISO/IEC 20922:2016 http://www.iso.org/iso/catalogue_detail.htm?csnumber=69466

[7] https://flume.apache.org/

parties or human). Retrospective security can be built on audit-data capture such as described in Section 5.

Retrospective verification of compliance (accountability) is particularly appropriate when: (1) detection of a violation occurs outside the computer system, such as when a user unexpectedly sees their personal data become publicly available; (2) when the violation occurs outside the systems that the subject directly controls; (3) when operators are highly trusted. Hospitals are a prime example of (3), where employees are highly accountable [66]. In an IoT scenario, an example of (1) and (2) could be devices disclosing information automatically in case of a detected/suspected emergency [6]; emergency disclosure should invariably be accompanied by a detailed audit record. In the case where someone is claiming compensation on the grounds that their data has been leaked by a system, audit can be used as a basis for evidence on whether a leak did or did not occur.

There are a few examples of analysis of provenance graphs to demonstrate conformance with certain regulations. Sakka et al. [57] discuss provenance in a cloud context in relation to document lifecycles. The application is banking under French regulation,[8] to ensure the probative value in court of electronic documents. This requires the emitter of any document to be identified and guarantees its integrity, which is achieved through provenance. Curbera et al. [19] proposed to use provenance to demonstrate compliance of businesses with regulation such as the Sarbanes-Oxley Act or the Health Insurance Portability and Accountability Act (HIPAA).

6.1 Identifying compliance violation

In [34], program dependence graphs (that can be considered a subset of provenance graphs) are analysed to demonstrate compliance with a given policy. For example, in a game where a user enters a number and the AI tries to guess this number, it can be demonstrated that the AI does not cheat if there is no dependency between the AI output and the user input. Similarly, the graph can be analysed to ensure that there is no dependency between a public output and a user password. Conversely, analyses of such graphs can reveal which information could be disclosed by a program [63], information that may in turn be disclosed to end-users. Similar analyses can be performed to demonstrate compliance with regulations, assuming such regulations can be expressed as constraints on information flow. In previous work [50], we used provenance to demonstrate compliance with the French data privacy agency guidelines in a cloud-connected smart home system.

In Section 3, we discussed the importance of the context in which an information transfer occurs. Verifying, for example, that information is disclosed only when a user is at a specific location, means verifying that there exists a dependency between the disclosure event and an item of data representing the location of the user (and obviously verifying that the location is correct). An absence of such a dependency means either (1) the location is not being verified by the application or (2) the location is inferred by other means (e.g. an action of a human operator on site, or point-to-point interaction with a fixed device). While a provenance graph is ideal for obtaining a comprehensive view of the context of an operation, extracting this context may require complex analysis and domain specific knowledge.

6.2 Legibility of the audit record

It is clear that regulators are aware of, but not yet resolved as to the solution to, legibility concerns in ensuring effective notice and control to users. So for example, the WP29 report on IoT elaborates its guidance to provide clear, comprehensible and user-friendly notices by suggesting a QR code or flashcode on objects themselves and, at the very least, requiring something more than general privacy policy on the data controllers' websites. It also emphasises that information should be offered to non-users whose personal data may be accessed within the IoT, as well as users. In separate WP29 reports, namely, the 2012 Opinion on Cloud Computing, and 2013 Opinion on Apps in Smart Systems, WP29 suggests what we are proposing in this paper: namely, clear audit trails so that end users can clearly see where their data is accessed and in what quantities. Audit is complemented in the WP29 reports by other tools that we should consider for future work, such as finding ways to allow easy modification of preferences without reducing control or inducing information fatigue. Another WP29 suggestion is for layered information notices, combined with meaningful icons to indicate certain data flows and uses.

Similarly, the FTC report on IoT discusses options for clear, prominent and conspicuous notice and choice, including developing video tutorials, affixing QR codes on devices, icons, offline communications, and providing choices at the point of sale, within set-up wizards, or in a privacy dashboard, command centre or management portal. Again, it emphasises not burying terms within lengthy documents. It also suggests the possibility of legislative or multi-stakeholder frameworks that could further refine permitted or prohibited uses. In earlier reports in March 2012 on Consumer Privacy[9] and in February 2013 on Mobile

[8] Code Civil Article 1316-1.

[9] https://www.ftc.gov/sites/default/files/documents/reports/federal-trade-commission-report-protecting-consumer-privacy-era-rapid-change-recommendations/120326privacyreport.pdf

Privacy and Transparency,[10] emphasis was given to standard notices, icons and other disclosures, with an extensive discussion, particularly in the latter, of different design concepts. Nevertheless, none of these reports tied legibility specifically to audit and data flows.

6.3 Structuring the audit graph

The partitioning of the notional global audit graph for IoT into intra and inter application context flows helps make the management of audit data more tractable, and more secure. Audit can focus on flows from a given context to ascertain that data has been suitably transformed or aggregated. Any policy violations can be investigated.

7 Conclusion

Data protection requirements will apply to significant volumes of data generated within the IoT. We have explored aspects of the legal and regulatory obligations that apply to the developers and deployers of IoT systems. Transparency and accountability lie at the heart of these obligations. Both require evidence (audit) of where data has flowed as an essential first step. Currently, such a capability is not provided or even considered at a technical level. As the IoT is consumer-led and its adoption requires trust by people and organisations, it is clear that means for improving transparency and accountability are very much needed.

This requires a great deal of work across diverse areas. Our focus is on how evidence of data flows can be gathered and queried. Other aspects are on how policy can be authored to align with law and regulation; how users can be assisted in expressing their wishes, as mandated by law; how all parties can be given transparency on what has happened to their data; how different sizes of organisation, e.g. SMEs, can be best supported to develop appropriate policy; and how the audit can be investigated to demonstrate compliance with law, thus achieving accountability.

We have previously argued that the ideal scenario for the future IoT is where law and regulation can be reflected in technically-enforceable policy. Ideally, law and regulation should be drafted with technical enforcement in mind. Technologists can but start from the assumption that this is possible, but it is a research issue in itself. This paper describes how compliance with such policy can be demonstrated by recording data provenance and auditing the flows of data throughout the IoT. That is, our focus here is not on the technology for policy enforcement, but on how the audit of data flows could assist in demonstrating compliance with (technical) policy, and by extension, with regulation. As a side-effect, the audit process would contribute to identifying discrepancies between law and expressed policy and contribute to the honing of the process.

Our goal is to make a strong case for audit in the IoT, without which there is little hope of transparency or accountability. With audit in place, evidence-based experience can be accumulated on what aspects of compliance can and cannot be demonstrated.

There are many challenges involved in providing such transparency and accountability for a system of the scale of the IoT. We have previously worked within a system structure of federated administrative domains, as a means of managing access control within and between domains. We see such a structure as helpful for managing the IoT, given interoperability standards. The ability to structure an audit graph and the ability to "zoom in" on certain contexts and subsystems, which are represented by a single node at a higher level, seems to us to be appropriate for the IoT. We outlined our own work in this area as a contribution towards improving transparency and accountability in the IoT through data provenance and audit.

Acknowledgements This work was supported by the US National Science Foundation under grant SSI-1450277 End-to-End Provenance, and the UK Engineering and Physical Sciences Research Council grant EP/K011510 CloudSafetyNet: End-to-end application security in the cloud. Cambridge authors acknowledge the support of Microsoft through the Microsoft Cloud Computing Research Centre.

References

1. Overview of the Internet of Things. Tech. Rep. (2012) Y.2060 ITU telecommunication standardization sector
2. Akoush S, Sohan R, Hopper A (2013) HadoopProv: towards Provenance as a First Class Citizen in MapReduce. In: Workshop on the theory and practice of provenance (TaPP'13). USENIX
3. Amir-Mohammadian S, Chong S, Skalka C (2016) Correct audit logging: theory and practice. In: International conference on principles of security and trust (POST'16). Springer
4. Armbrust M, Das T, Davidson A, Ghodsi A, Or A, Rosen J, Stoica I, Wendell P, Xin R, Zaharia M (2015) Scaling Spark in the real world: performance and usability. International Conference on Very Large Data Bases (VLDB) 8(12):1840–1843
5. Bacon J, Moody K (2002) Toward open, secure, widely distributed services. Commun ACM 45(6):59–64
6. Bacon J, Singh J, Trossen D, Pavel D, Vastardis N, Yang AB, Pennington K, Clarke S, Jones SG (2012) Personal and social communication services for health and lifestyle monitoring. In: Proceedings 1st international conference on global health challenges (Global Health 2012), with IARIA Datasys, Venice, p 2012
7. Barbieri DF, Braga D, Ceri S, VALLE ED, Grossniklaus M (2010) C-SPARQL: a continuous query language for RDF data streams. Int J Semantic Comput 4(01):3–25
8. Bates A, Butler K, Moyer T (2015) Take only what you need: leveraging mandatory access control policy to reduce provenance storage costs. In: Workshop on theory and practice of provenance. USENIX, pp 7–7

[10] https://www.ftc.gov/sites/default/files/documents/reports/mobile-privacy-disclosures-building-trust-through-transparency-federal-trade-commission-staff-report/130201mobileprivacyreport.pdf

9. Bates A, Mood B, Valafar M, Butler K (2013) Towards secure provenance-based access control in cloud environments. In: Conference on data and application security and privacy. ACM, pp 277–284
10. Bates A, Tian D, Butler K, Moyer T (2015) Trustworthy whole-system provenance for the Linux kernel. In: Security symposium. USENIX
11. Braun U, Garfinkel S, Holland DA, Muniswamy-Reddy KK, Seltzer MI (2006) Issues in automatic provenance collection. In: Provenance and annotation of data. Springer, pp 171–183
12. Braun U, Shinnar A, Seltzer MI (2008) Securing provenance. In: Summit on hot topics in security (HotSec'08). USENIX
13. Cadenhead T, Khadilkar V, Kantarcioglu M, Thuraisingham B (2011) A language for provenance access control. In: Conference on data and application security and privacy. ACM, pp 133–144
14. Carata L, Akoush S, Balakrishnan N, Bytheway T, Sohan R, Selter M, Hopper A (2014) A primer on provenance. Commun ACM 57(5):52–60
15. Chaudhry A, Crowcroft J, Howard H, Madhavapeddy A, Mortier R, Haddadi H, McAuley D (2015) Personal data: thinking inside the box. In: Proceedings of the fifth decennial Aarhus conference on critical alternatives. Aarhus University Press, pp 29–32
16. Chen P, Evans T, Plale B (2016) Analysis of memory constrained live provenance. In: International provenance and annotation workshop. Springer, pp 42–54
17. Coker G, Guttman J, Loscocco P, Herzog A, Millen J, O'Hanlon B, Ramsdell J, Segall A, Sheehy J, Sniffen B (2011) Principles of remote attestation. Int J Inf Secur 10(2):63–81
18. Crawl D, Wang J, Altintas I (2011) Provenance for mapreduce-based data-intensive workflows. In: Workshop on workflows in support of large-scale science. ACM, pp 21–30
19. Curbera F, Doganata Y, Martens A, Mukhi NK, Slominski A (2008) Business provenance–a technology to increase traceability of end-to-end operations. In: On the move to meaningful internet systems: OTM 2008. Springer, pp 100–119
20. Edwards A, Jaeger T, Zhang X (2002) Runtime verification of authorization hook placement for the Linux security modules framework. In: Conference on computer and communications security (CCS). ACM, pp 225–234
21. Flittner M, Balaban S, Bless R (2016) Cloudinspector: A transparency-as-a-service solution for legal issues in cloud computing. In: IC2E international workshop on legal and technical issues in cloud computing (CLaw'16). IEEE
22. Fu Q, Zhu J, Hu W, Lou JG, Ding R, Lin Q, Zhang D, Xie T (2014) Where do developers log? an empirical study on logging practices in industry. In: International conference on software engineering (ICSE). ACM, pp 24–33
23. Ganapathy V, Jaeger T, Jha S (2005) Automatic placement of authorization hooks in the Linux security modules framework. In: Conference on computer and communications security (CCS). ACM, pp 330–339
24. Gehani A, Tariq D (2012) Spade: Support for provenance auditing in distributed environments. In: Middleware conference. IEEE/ACM/IFP/USENIX, pp 101–120
25. Gonzalez JE, Low Y, Gu H, Bickson D, Guestrin C (2012) Powergraph: distributed graph-parallel computation on natural graphs. In: Symposium on operating systems design and implementation (OSDI'12). USENIX, p 2
26. Gonzalez JE, Xin RS, Dave A, Crankshaw D, Franklin MJ, Stoica I (2014) Graphx: graph processing in a distributed dataflow framework. In: Symposium on operating systems design and implementation (OSDI'14), vol 14, pp 599–613
27. Gubbi J, Buyya R, Marusic S, Palaniswami M (2013) Internet of Things (IoT): A vision, architectural elements, and future directions. Futur Gener Comput Syst 29(7):1645–1660
28. Hayton RJ, Bacon JM, Moody K (1998) Access control in an open distributed environment. In: 1998 IEEE symposium on security and privacy, 1998. Proceedings. IEEE, pp 3–14
29. Hon WK, Millard C, Singh J (2016) Twenty legal considerations for Clouds of Things. Queen Mary School of Law Legal Studies Research Paper (216)
30. Hussein J, Moreau L, Sassone V (2015) Obscuring provenance confidential information via graph transformation. In: IFIP International conference on trust management. Springer, pp 109–125
31. Interlandi M, Shah K, Tetali SD, Gulzar MA, Yoo S, Kim M, Millstein T, Condie T (2015) Titian: data provenance support in Spark. Conference on Very Large Databases (VLDB'15) 9(3): 216–227
32. Jaeger T, Edwards A, Zhang X (2004) Consistency analysis of authorization hook placement in the Linux security modules framework. ACM Trans Inf Syst Secur (TISSEC) 7(2):175–205
33. Jaeger T, Sailer R, Shankar U (2006) PRIMA: Policy-reduced integrity measurement architecture. In: ACM Symposium on access control models and technologies (SACMAT). ACM, pp 19–28
34. Johnson A, Waye L, Moore S, Chong S (2015) Exploring and enforcing security guarantees via program dependence graphs. In: ACM SIGPLAN notices, vol 50. ACM, pp 291–302
35. Kemmerer RA, Vigna G (2002) Intrusion detection: a brief history and overview. IEEE Computer 35(4):27–30
36. Keoh SL, Kumar S, Tschofenig H (2014) Securing the internet of things: a standardization perspective. Internet of Things Journal 1(3):265–275
37. Kohnstamm J, Madhub D (2014) Mauritius Declaration on the Internet of Things. In: International conference of data protection and privacy commissioners
38. Kyrola A, Blelloch GE, Guestrin C et al. (2012) GraphChi: large-scale graph computation on just a PC. In: Symposium on operating systems design and implementation (OSDI'12), vol 12. USENIX, pp 31–46
39. Lampson BW (2004) Computer security in the real world. IEEE Computer 37(6):37–46
40. Macko P, Seltzer M (2012) A general-purpose provenance library. In: Workshop on the theory and practice of provenance (TaPP'12). Usenix
41. McKinsey Global Institute (2015) The Internet of Things: mapping the value beyond the hype
42. Mineraud J, Mazhelis O, Su X, Tarkoma S (2016) A gap analysis of Internet-of-Things platforms. Comput Commun ACM 89(C): 5–16
43. Missier P, Belhajjame K, Cheney J (2013) The W3C PROV family of specifications for modelling provenance metadata. In: Conference on extending database technology (EDBT). ACM, pp 773–776
44. Missier P, Bryans J, Gamble C, Curcin V, Danger R (2014) Provabs: model, policy, and tooling for abstracting prov graphs. In: International provenance and annotation workshop. Springer, pp 3–15
45. Moyer T, Gadepally V (2016) High-throughput ingest of data provenance records into Accumulo. In: High performance extreme computing conference (HPEC). IEEE, pp 1–6
46. Neumann T, Weikum G (2010) The RDF-3x engine for scalable management of RDF data. VLDB J 19(1):91–113
47. Ni Q, Xu S, Bertino E, Sandhu R, Han W (2009) An access control language for a general provenance model. In: Workshop on secure data management. Springer, pp 68–88
48. Park J, Nguyen D, Sandhu R (2012) A provenance-based access control model. In: Annual international conference on privacy, security and trust. IEEE, pp 137–144
49. Pasquier T (2017) Camflow/camflow-dev: v0.3.0. doi:10.5281/zenodo.571427. https://github.com/CamFlow/camflow-dev

50. Pasquier T, Eyers D (2016) Information flow audit for transparency and compliance in the handling of personal data. In: IC2E international workshop on legal and technical issues in cloud computing (CLaw'16). IEEE
51. Pasquier T, Singh J, Bacon J, Eyers D (2016) Information Flow Audit for PaaS clouds. In: International conference on cloud engineering (IC2E). IEEE
52. Pasquier T, Singh J, Eyers D, Bacon J (2015) CamFlow: managed data-sharing for cloud services. IEEE Trans Cloud Comput (TCC)
53. Pohly DJ, McLaughlin S, McDaniel P, Butler K (2012) Hi-fi: collecting high-fidelity whole-system provenance. In: Annual computer security applications conference. ACM, pp 259–268
54. Povey D (1999) Optimistic security: a new access control paradigm. In: Proceedings of the 1999 workshop on new security paradigms. ACM, pp 40–45
55. Roy A, Mihailovic I, Zwaenepoel W (2013) X-stream: edge-centric graph processing using streaming partitions. In: Proceedings of the twenty-fourth ACM symposium on operating systems principles (SOSP). ACM, pp 472–488
56. Sailer R, Zhang X, Jaeger T, Van Doorn L (2004) Design and implementation of a TCG-based integrity measurement architecture. In: USENIX Security symposium, vol 13. USENIX, pp 223–238
57. Sakka MA, Defude B, Tellez J (2010) Document provenance in the cloud: constraints and challenges. In: Networked services and applications-engineering, control and management. Springer, pp 107–117
58. Singh J, Pasquier T, Bacon J, Ko H, Eyers D (2016) Twenty security considerations for cloud-supported Internet of Things. IEEE Internet of Things Journal 3(3):269–284
59. Singh J, Pasquier T, Bacon J, Powles J, Diaconu R, Eyers D (2016) Big ideas paper: policy-driven middleware for a legally compliant internet of things. In: ACM/IFIP/USENIX middleware. ACM
60. Smith M, Szongott C, Henne B, von Voigt G (2012) Big data privacy issues in public social media. In: 2012 6th IEEE international conference on digital ecosystems and technologies (DEST). IEEE, pp 1–6
61. Stolfo SJ, Salem MB, Keromytis AD (2012) Fog computing: mitigating insider data theft attacks in the cloud. In: 2012 IEEE symposium on security and privacy workshops (SPW). IEEE, pp 125–128
62. Takabi H, Joshi J, Ahn G (2010) Security and privacy challenges in cloud computing environments. IEEE Secur Priv 8(6):54–57
63. Vaughan JA, Chong S (2011) Inference of expressive declassification policies. In: 2011 IEEE Symposium on security and privacy. IEEE, pp 180–195
64. Weber RH (2010) Internet of Things–new security and privacy challenges. Computer Law & Security Review 26(1):23–30
65. Weitzner DJ (2007) Beyond secrecy: new privacy protection strategies for open information spaces. IEEE Internet Comput 11(5):96–95
66. Weitzner DJ, Abelson H, Berners-Lee T, Feigenbaum J, Hendler J, Sussman GJ (2008) Information accountability. Commun ACM 51(6):82–87
67. Xie Y, Muniswamy-Reddy KK, Feng D, Li Y, Long DD (2013) Evaluation of a hybrid approach for efficient provenance storage. ACM Transactions on Storage (TOS) 9(4):14
68. Xie Y, Muniswamy-Reddy KK, Long DD, Amer A, Feng D, Tan Z (2011) Compressing provenance graphs. In: Workshop on the theory and practice of provenance (TaPP'11). Usenix
69. Zhu X, Chen W, Zheng W, Ma X (2016) Gemini: a computation centric distributed graph processing system. In: Symposium on operating systems design and implementation (OSDI). USENIX
70. Ziegeldorf JH, Morchon OG, Wehrle K (2014) Privacy in the Internet of Things: threats and challenges. Secur Commun Netw 7(12):2728–2742