Twenty Security Considerations for Cloud-Supported Internet of Things

Jatinder Singh, Thomas Pasquier, Jean Bacon, Hajoon Ko, and David Eyers

***Abstract*—To realize the broad vision of pervasive computing, underpinned by the "Internet of Things" (IoT), it is essential to break down application and technology-based silos and support broad connectivity and data sharing; the cloud being a natural enabler. Work in IoT tends toward the subsystem, often focusing on particular technical concerns or application domains, before offloading data to the cloud. As such, there has been little regard given to the security, privacy, and personal safety risks that arise beyond these subsystems; i.e., from the wide-scale, cross-platform openness that cloud services bring to IoT. In this paper, we focus on security considerations for IoT from the perspectives of cloud tenants, end-users, and cloud providers, in the context of wide-scale IoT proliferation, working across the range of IoT technologies (be they things or entire IoT subsystems). Our contribution is to analyze the current state of cloud-supported IoT to make explicit the security considerations that require further work.**

***Index Terms*—Cloud, compliance, data, Internet of Things (IoT), law, privacy, security.**

I. Introduction

During the last decades of the Twentieth Century, there was much research into sensor and communications technologies. At that time, sensor-based systems tended to be developed in "silos," being localized, application- and technology-specific. It became evident that sensor data could potentially be used for many diverse purposes if a means of sharing could be devised. The term "Internet of Things" (IoT), first coined in 1999 by Ashton at MIT,[1] came to be used to capture this aspiration: 1) based on ever-wider connectivity of sensor/actuator-based systems, more general data sharing would become possible than within the specific applications for which those systems were developed and 2) computers would become autonomous, able to collect data and take decisions based on them, without human intervention. Moreover, IoT represents a broader move to the vision of pervasive or ubiquitous computing [1].

Manuscript received December 01, 2014; revised April 15, 2015 and May 26, 2015; accepted July 06, 2015. Date of publication July 23, 2015; date of current version May 10, 2016. This work was supported by U.K. Engineering and Physical Sciences Research Council under Grant EP/K011510 CloudSafetyNet: End-to-End Application Security in the Cloud and Microsoft through the Microsoft Cloud Computing Research Centre.

J. Singh, T. Pasquier, J. Bacon, and H. Ko are with the Computer Laboratory, University of Cambridge, Cambridge CB3 0FD, U.K. (e-mail: jatinder.singh@cl.cam.ac.uk).

D. Eyers is with the Department of Computer Science, University of Otago, Dunedin 9016, New Zealand.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JIOT.2015.2460333

[1][Online]. Available: http://www.rfidjournal.com/articles/view?4986, accessed on Apr. 2015.

Recently, the IoT concept has captured imaginations within government and commerce, as a technology capable of supporting immense growth [2], [3]. However, systems aiming at this wider vision are in their infancy. Sensor/actuator-based systems have been developed independently of the IoT vision of open data sharing. It is crucial that the security, privacy, and personal safety risks arising from open access to data, across and beyond these systems, are evaluated and addressed.

IoT potentially covers a wide range of applications, including smart home systems, smart street lighting, traffic congestion detection and control, noise monitoring, city-wide waste management, real-time vehicle networks, and smart city frameworks [4]. At the individual level, personal health and lifestyle monitoring systems are being integrated with general healthcare services [5]. Such application scenarios tend to be sensor/actuator-based, each developed for a single purpose. In contrast, the IoT philosophy is the wide-scale integration of potentially all technology, including individual devices, applications, servers, and so forth, in addition to sensors/actuators, i.e., the data from a range of different sources is capable of diverse potential application and should be developed with broad usage and wide availability in mind.

The cloud is an obvious technology for achieving this open sharing. Cloud computing has evolved to manage, process, and store *big data*, that, e.g., has arisen from services such as search engines. Data analytics has become an essential complement to cloud-hosted web services. Similar services can be used for large-scale data from IoT systems (including those that are mobile), making them independently shareable and widely available.

The cloud is an ideal component in an IoT architecture. First, because cloud services can operate across a range of systems, services, and devices, it provides the natural point for: 1) data aggregation and analysis and 2) the management, control, and coordination of the range of systems and services. Furthermore, (3) cloud services offer benefits in terms of resource management, as clouds are always ON, can scale to meet demand, and can allow the offloading from constrained hardware of data (for computation [6] and storage) and management specifics. In this paper, we use *IoT-cloud* to refer to IoT architecture that incorporates cloud services. Fig. 1 illustrates a variety of IoT applications, supported by cloud services.

Any closed subsystem (see Section II-B), e.g., which might represent a low-level sensor network, or a group of devices behind a firewall/access-point is assumed to have a gateway (a.k.a. edge-server or hub). The support for connectivity and open sharing via cloud services allows, e.g., emergency services to interact with traffic control, power (utility) providers,

2327-4662 © 2015 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See http://www.ieee.org/publications_standards/publications/rights/index.html for more information.

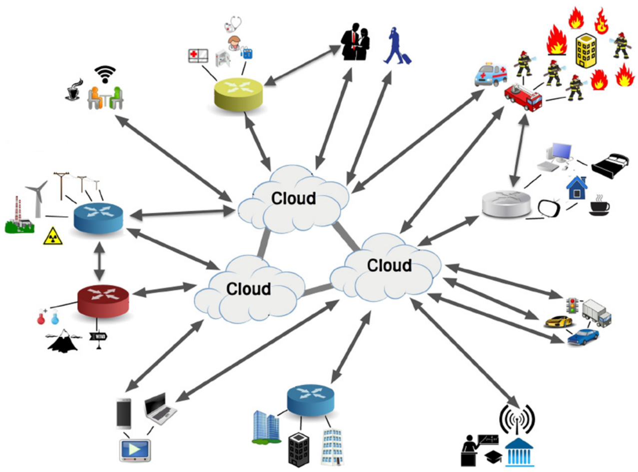


Fig. 1. Illustration of IoT including cloud services (IoT-cloud).

home monitoring, and the ambulance service and hospitals, as appropriate. An IoT architecture allows *different applications* to be built using the same set of sensors, actuators, and devices.

Many IoT-enabled services are developed with a single application in mind, with little consideration of security issues beyond local concerns, e.g., security might exist within a sensor network, but not when data is passed outside. Furthermore, IoT applications are linked to the physical world and can directly influence and change it. For example, Leverett [7] discusses systems that were believed to be within a secured network but were in fact directly accessible through the Internet. They were often poorly protected, e.g., through a simple password scheme, or sometimes not at all. Ventilation and temperature management systems for hospitals were compromised, putting patients' lives at risk, through error or deliberate attack.

As the number of connected devices increases and their usage becomes an important part of everyday life, security, privacy, and personal safety issues will arise.

Security concerns are already seen to inhibit the uptake of cloud services, especially by public bodies with responsibility for sensitive data, such as healthcare services [8]. Similar concerns arise for IoT-cloud, exacerbated by the sheer scale of IoT. As the number of sources/sinks increases, managing and securing these appropriately becomes a challenge. Privacy is also a real concern—personal data could be collected from a wide range of sources. Benign sources and "anonymized" data may reveal little in isolation, but combining data from a number of sources can result in privacy-invading inference [9]. A wider challenge for IoT is a fuller awareness of the possible consequences of open connectivity.

This paper focuses on the security considerations for IoT-cloud, given that cloud services act as "glue" that can integrate and mediate "things," as well as provide data processing, storage, and management for individual "things." In this context, we analyze the current state of cloud service offerings for IoT and consider their security provision from the perspectives of cloud tenants, end-users, and cloud providers, focusing on the interplay between them.

The core contribution of this paper is to identify a range of security concerns specific to IoT's use of cloud services: we present *20 key security considerations* for IoT-cloud. Each section operates to encapsulate a number of considerations, which are summarized in Table I.

We first consider issues accessing the cloud (Section III), exploring issues of secure transport (1) and cloud access controls (2), before considering the range of data management concerns in Section IV, Considerations 3–8. We then discuss issues of identity management (Section V, Considerations 9 and 10) followed by the issues of scale (Section VI, Considerations 11 and 12) that are inherent to IoT-cloud. Dealing with malicious "things" and associated attacks is explored in Section VII, Considerations 13 and 14. The focus then turns to the integrity of cloud services, considering certification and trustworthiness (Section VIII, Considerations 15 and 16), and related issues of compliance, transparency, and responsibility in Section IX, Considerations 17–19. Finally, in Section X, we survey the emerging directions in cloud computing, including fog, edge, and decentralized clouds, and examine the associated IoT security concerns (Consideration 20).

Section XI summarizes the considerations in Table I with respect to the confidentiality, integrity, and availability (CIA) security properties, in each case, indicating where "off-the-shelf" security mechanisms can be used (green), where additional but tractable work is needed (amber) and where significant research is required (red).

We now begin by providing background and establishing the context for these considerations.

II. Architecture Overview: Clouds and Things

A. Cloud-Computing Terminology

Advances in networking, bandwidth, resource management, and virtualization technologies have resulted in service models that involve provisioning computing-as-a-service. Cloud computing, "the cloud," involves *cloud service providers* (providers): those offering the service, provisioning, and managing a set of technical resources; among *tenants*: those consuming the cloud services through direct relationships with providers. The providers' business model is generally to leverage economies of scale by sharing resources between tenants, while tenants gain from being able to pay only for the resources they require, thus removing a costly start-up base and being able to acquire service elasticity—to rapidly scale up and/or scale down resources in response to fluctuations in demand—and more generally, improving access to storage and computational services. The *end-user* of a system may interact with a cloud provider either directly or indirectly via tenant-provided services.

Cloud service offerings are generally divided into three main categories: 1) infrastructure as a service (*IaaS*); 2) platform as a service (*PaaS*); and 3) software as a service (*SaaS*) as shown in Fig. 2. In IaaS, the cloud service provider is responsible for the management of the network, hardware, and hypervisor. PaaS service providers offer, in addition, the managed OS and application environment. SaaS service providers manage everything on behalf of the tenants, including the application. There are other categories emerging, including network as a service, brokers as a service, sensors as a service, etc., though

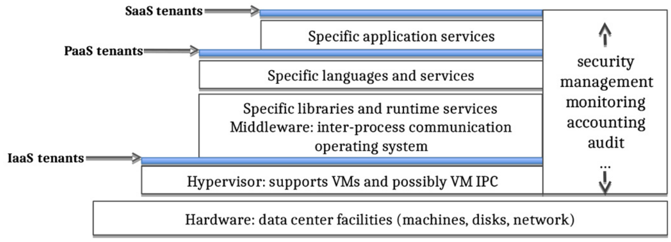


Fig. 2. Cloud service provision architectural overview [10]. For each service model, a tenant is provided all that below the blue line.

most of these focus on offering a particular component in cloud service composition, rather than a larger scale platform.

Another common division of cloud systems is *private* and *public*. Public clouds are the most common, where the cloud provider shares resources (hardware or perhaps software such as databases) between tenants. *Virtual machines* (VMs) or *containers* are used to ensure a separation between tenants and their resources. The public cloud brings benefits in terms of economies of scale, and is of most interest and with most potential, moving forward. In a private cloud model, the tenant is offered a dedicated (unshared) set of resources. This is analogous to "in-house" management, giving the tenant greater control and an increased sense of security. *Hybrid* clouds bridge the two, where some resources (e.g., potentially sensitive data) might be processed in a private cloud, others on the public cloud. Data and processing may be transferred between the two, when and where appropriate, e.g., for scaling and analytics.

For example, in the U.K., there is a National Cancer Registration Service (NCRS)[2] that holds cancer-related health records in a private datacenter to comply with national regulations on safeguarding patient confidentiality. Patients can see their own data, but only through a public web portal. The NCRS makes datasets available for medical research, but given the sensitive, personal nature of the data, it must be anonymized before leaving the private cloud. Strong audit is required to manage the anonymization and data migration processes.

B. IoT-Cloud Components

The term the IoT is broad, often used in a number of technical contexts to focus on very specific concerns, such as wireless (radio) communication aspects, sensor networks, machine-to-machine (M2M) communication, human/environmental and technical interactions, and so forth. For the purposes of this paper, we consider IoT in terms of supporting the wider vision of pervasive/ubiquitous computing, whereby the whole range of sensors, devices, applications, systems, servers, clouds (i.e., anything) has the potential to interact in order to realize some functionality.[3]

We refer to IoT *subsystems* in order to represent a closed and/or self-contained network of "things." These subsystems generally have a *gateway* component (a.k.a. hub or edge-server) with the functionality of masking heterogeneity and controlling the data flowing in/out, and in some cases, mediating the "things." Subsystems may be application-centric networks, either fixed or rapidly instantiated and/or temporary (*ad hoc*) in nature, e.g., supporting a smart home, emergency services during a catastrophe; or those comprising a particular technology domain (ecosystem), such as a proprietary sensor network, or control system in an industrial assembly line.

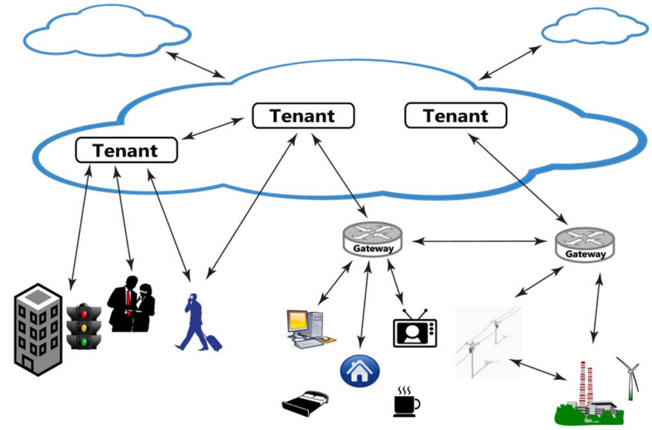


Fig. 3. Illustration of the interactions within an IoT-cloud.

To facilitate a wide-ranging discussion, in this paper, we consider a *thing* as any entity, physical, or virtual, capable of *interaction* (data exchange) [11]. Our focus is particularly on "things" interacting with cloud services. Some "things" will be individual items such as Internet-enabled video cameras and fridges. A subsystem is also considered a "thing," because the cloud provider sees and interacts (only) with the subnetwork's gateway component; the gateway represents the end-point of the cloud interaction, mediating between the subsystem and the cloud. In line with this definition, it is possible that a single device could be considered, from the cloud perspective, as several "things." For example, a smartphone has the potential to host a range of different applications, each of which are capable of direct cloud interaction, but these would act (separately) as gateways for the data collected by the phone's sensors. Fig. 3 depicts the interaction of clouds and "things," including those through IoT subsystems.

As introduced in Section I, "things" are typically developed for particular applications, or within technical domains (in terms of radio, communication protocols, APIs, etc.). Therefore, in practice, most of the security engineering focus is on protection against specific, targeted attacks, with little consideration given to security issues beyond these domains. However, harnessing the full potential of IoT involves using/reusing system components, when and where appropriate, to realize new, possibly previously unforeseen functionality and services. Hence, this paper focuses on a (mostly overlooked) area by considering security with respect to the interplay of "things" and their interactions with cloud services.

C. Leveraging the Cloud for IoT

In Section I, we presented a historical IoT perspective, where various communication and lower level sensor management

[2][Online]. Available: http://www.ncr.nhs.uk, accessed on Apr. 2015.
[3]Of course, this is only the *potential*. There are practical limits in terms of application/network boundaries, economic and ownership considerations, etc.

networks were developed for specific purposes. Early work in such areas often mentioned offloading computing or data onto a "server." Moving forward, we saw "server" being replaced by "cloud," and we now see many IoT solutions as tightly integrated with cloud services. For instance, a recent survey showed that among the 38 IoT platforms surveyed, 33 relied on cloud or other centralized services [12].

There are good reasons for using cloud services to support IoT, in terms of general resourcing.

1) Cloud services are "always ON," and globally accessible, so "things" can be located anywhere, be mobile, can transmit different data at different times.
2) Cloud services are built to scale rapidly, which ideally suits IoT in which many "things" can communicate at different data rates, and at different times.
3) They help manage resource constraints. Many "things" will be limited in terms of computational power, battery, storage capacity, etc. The ability to shift some of this load to the cloud helps to alleviate these limitations.

Furthermore, cloud services can easily *operate across a range of "things."* Cloud services can be used to mediate between different "things," to enable: 1) wide-ranging data sharing and 2) to manage and control a range of different "things" as appropriate. Therefore, using the cloud to support IoT naturally provides *cross-"thing" management*. It enables data and control flows (e.g., coordination policy) to move *horizontally*, working across a wide-range of "things." This is crucial to the wider vision of IoT, enabling pervasive computing more generally. Indeed, this leads to *big data*-proper, by providing the means for personalization, customization, and automated/intelligent actions across a range of different applications, "things," and physical environments [13].

1) IoT-Cloud Interactions: We are primarily concerned with the interactions between "things" and the cloud, see Fig. 3.

The cloud provider offers services and infrastructure for data storage, computation, etc. The service model could be IaaS, PaaS, or SaaS, depending on the specific service offerings and requirements. For IaaS or PaaS services, tenants might be applications serving a number of different "things," and therefore end-users might also be tenants, e.g., for a user's "quantified self" data. The considerations we explore typically apply across cloud service models.

As discussed above, in order to leverage the full power of the IoT, we need the possibility for data to be shared across a range of applications, i.e., horizontally, between "things." This has not been the vision of many existing IoT systems, developed as one-off services in a closed and/or limited application space (i.e., a vertical silo). *Nor was interapplication sharing the vision of cloud service providers*; strong isolation between applications was their prime concern, achieved through tenant isolation technologies (e.g., VMs and containers).

A motivating example representative of the wider vision is the concept of a *smart city*, where data is generated on local weather conditions, car park occupancy, traffic congestion, bus location, pollution, building usage, etc. The vision is for this wide range of data to be made available, via cloud-hosted platforms, for services (public or private) to be built on top. For this, data will need to be shared, and analytics operate over a range of data sources, repositories, and "things." For instance, there may be services that analyze data in combination and issue emergency alerts, e.g., when weather conditions are extreme and/or traffic is congested, and emergency services must be routed to an accident. We discuss this further in Section IV.

2) Scope of IoT-Cloud Security Considerations: Security remains an ongoing challenge for systems generally. Indeed, there are many problems to be solved in the IoT world [14]—i.e., within the IoT subsystems that we have mentioned—including network protocols, radio management, standardization, internal security, and privacy [15]. Similarly, there are ongoing security issues for cloud services [16], many of which are concerned with provider trust.

The focus of this paper differs as it specifically explores security at the level where "things" and the cloud interact. We do not consider lower level, subsystem-specific security aspects/attacks, and further, only consider general cloud security issues when relevant to the IoT-cloud. Through the rest of this paper, we introduce security considerations that should be taken into account when cloud services are integrated with IoT, and in Section XI highlight those considerations that require further research.

III. Accessing the Cloud

Communication underpins the interactions between "things" and the cloud. There is a bidirectional flow of information. Data might flow from "things" to the cloud, perhaps for storage or analytics. The cloud may also be the mediator and/or conduit through which data (including actuating commands) is sent to "things." Much data will be sensitive, whether alone or in aggregate. It is, therefore, important that communication is secure, and user-access to cloud services is properly controlled.

Consideration 1: Secure Communications. There are two motivations for securing communication: 1) secrecy: preventing eavesdropping and data leakage and 2) integrity: protecting data from corruption/interference. Note that here we do not consider communication within subsystems, but rather are concerned with the interaction of "things" with cloud services.

Secure communication is required to prevent unauthorized access to data (or metadata) that might be sensitive. Transport layer security (TLS) [17] uses cryptography to establish a secure channel to protect transmissions (including metadata such as protocol state, thus limiting side-channels) from both eavesdropping and interference. TLS employs a certificate-based model, relying on public key infrastructure (PKI) and certificate authorities for authentication.

TLS is a common feature of cloud-provider offerings, and can be used to secure the confidentiality and integrity of communications between "things" and the cloud provider.

With a general view to making secure communication more commonplace, there is recent work on enabling TLS over protocol stacks other than TCP/IP to better suit the requirements of "things," in terms of complexity and resource requirements. Examples include datagram transport layer security (DTLS [18], [19]) for datagram oriented protocols such as User Datagram Protocol (UDP), and LLCPS [20] that applies TLS over the near-field communications Logical Link Control

Protocol (LLCP). Depending on the deployment, architecture, and interfaces to cloud services, these technologies could facilitate new forms of secure "thing"–cloud interactions.

Apart from TLS, there are, of course, other mechanisms of securing "thing"–cloud communication. Data can be encrypted by applications, which protects data not only in transit but also beyond. Sharing secrets naturally entails management and engineering considerations [21]. We explore this aspect in Consideration 7.

Aside from any vulnerabilities inherent in the approach, the protection offered by any secure communication mechanism is only as good as its implementation. For example, the recent Heartbleed vulnerability in the widely used OpenSSL library is estimated to have left 24%–55% of TLS protected endpoints open to attack [22]. Extra care and consideration must be given to the newer schemes and implementations currently being developed to support IoT, especially those that may not have been widely scrutinized or deployed.

Consideration 2: Access Controls for IoT-Cloud. It is important that (external) access to cloud resources is regulated. *Access controls* [21] operate to govern the actions that may be taken on objects, be they accessing particular data (a file, record, and data stream), issuing a query, performing some computation, and so forth. Controls are typically *principal* focused, in the sense that control policy governing a particular action is defined to regulate those undertaking the action, enforced when they attempt to take that action.

There are two aspects to access control: 1) *authentication* and 2) *authorization*. Authentication refers to verifying who a principal is, i.e., are they who they say they are? Authorization rules follow authentication; once a principal is identified, what are their rights and privileges; what actions are they authorized to undertake?

In a general cloud context, the provider will offer access controls to ensure that only the correct tenants/users (the *principals*) access the appropriate data and services. Cloud providers often have login/credential-based services for authenticating tenants/users. Authorization policy will be enforced as a principal attempts to take an action, based on their level of privilege, which might allow them to access storage and files held by the provider, initiate computation services, etc. The precise controls will depend on the specific service offering, but often include *access control lists*, *role-based access controls*, *capabilities*, etc. See [21] for an overview of a number of security engineering techniques.

In an IoT context, a challenge for any access control regime is accounting for the fact that the interactions between "things" may involve encounters with "things" never before seen, or owned and operated by others. Toward this, *trusted platform modules* (TPM) [23] offer promise by providing strong guarantees, e.g., with respect to device identity [24] and configuration [25], which access control mechanisms can leverage.

IoT-cloud poses extra challenges. The first concerns authentication, given the size and scale of the IoT vision, correctly identifying the "things" and determining the relevant cloud services/tenant applications is a real concern; Section V is dedicated to issues of identity. There are also difficulties in the fact that infrastructure and data may be shared. Currently, cloud policy is focused: authorization rules are to ensure that a tenant accesses only its own resources, i.e., their files, VMs, databases, etc. However, for the IoT-cloud, the lines are blurred. The data and resources of a tenant may be relevant to a number of different principals, and/or may control and coordinate a number of "things." Policy must be able to be consistently defined and applied across both of these dimensions.

Access controls may be contextual, e.g., people may in general only access data concerning themselves. In exceptional circumstances, such as medical emergencies [26], wider access may be desirable, as specified by "break-glass policies." Mechanisms are required to enable flexible access control policies to be defined by different parties, while also being able to identify and resolve potential policy conflicts. Such concerns are nontrivial, and will likely require some external constraints, such as ownership or economic incentives (e.g., those paying for the service) to help make access control policy more manageable.

Note that access controls govern the tenant/user–provider interactions at the interface between them. These mechanisms typically do not, by themselves, offer control beyond that point, e.g., how their data is managed internally by the provider(s) (see Consideration 6).

Controlling and coordinating "things": The cloud will play a role in mediating and coordinating "things," where actuating commands, the initiation/cessation of data flows, and so forth will be initiated from the cloud. It is clear that "things" will need to maintain some form of access control, to prevent potentially anyone from taking over. This is illustrated, e.g., by an access control vulnerability discovered in a consumer lighting system, allowing an attacker to issue lighting commands (causing blackout) by masquerading as a user-device [27].

The role of the cloud as a mediator of "things," brings several considerations. First, the access controls are not necessarily symmetric, in that the process by which a "thing" may access the cloud is not necessarily the same as how the cloud can initiate access to the "thing." Because there will be far more "things" than cloud services, there will likely also be a far greater range of access control implementations, credential services, etc., employed by "things." The cloud provider must be able to account for these. As such, standardization is clearly an important issue, and the role of gateway components will assist in limiting the diversity.

Second, any cloud-based mediation and coordination will be driven by policy components, many of which reside within the cloud. To realize the wider IoT vision, policy enforcement mechanisms must be sufficiently flexible to be defined across the range of devices, while accounting for the differences in access control models, i.e., the cloud-deployed policy enforcement components must be able to dynamically switch between them to enable context-aware coordination when/where appropriate, e.g., to adapt security levels based on a perceived risk [28].

Care must also be taken to ensure that coordination policy does not lead to further vulnerabilities—see Consideration 18 and [27] for a practical demonstration.

IV. Data Management in the Cloud

The IoT-cloud is such that "things" upload their data to the cloud, the provider offering various storage, computational, or other services. In addition, the cloud is the natural location for policy enforcement that has broader scope than a single "thing"; in terms of affecting a range of "things" or due to external changes in context. The cloud provider becomes responsible,[4] to some degree, for managing and acting on the data it holds and processes, regardless of the service level (IaaS, PaaS, or SaaS) [95].

We have considered the interactions between cloud services and "things," regarding secure data transmission, and how access to resources can be regulated. We now explore security considerations regarding data management within the cloud.

Consideration 3: Identifying Sensitive Data. In an IoT context, it will often be the case that data is considered sensitive. This is because data will encapsulate various aspects of the physical environment, including highly personal information about individuals, groups, and companies, and can also have physical consequences, e.g., actuating commands.

It is, therefore, important that security mechanisms are designed to take account of the potential sensitivity of the data. A recent example illustrating a failure of such involved a baby monitor, where an iOS device on the local network could listen in without being subjected to access controls [30]. Furthermore, any device that had ever accessed the monitor could then *remotely* listen in, anytime, anywhere. This represents a clear failure to recognize and/or account for the sensitivity of the audio feed.

Identifying the "thing" that produces data may not always be sufficient to determine how sensitive its data. For example, a location sensor may be considered as generating sensitive data when representing the movements of a particular person, but the data produced by the same sensor may be less sensitive when it is attached to freight in transit. Furthermore, sometimes, only specific items/data-instances are highly sensitive, even when produced by the same "thing," e.g., a facial recognition device in a public space could provide the current location of the Prime Minister, thus having national security implications.

Note also that the combination of data can raise the level of sensitivity: we explore this in Consideration 8.

Consideration 4: Cloud Architectures: Public, Private, or Hybrid? Where particularly sensitive, there may be decisions to prevent data being placed on a public cloud [31], as is the case for health records in general or some specific category such as cancer records [32]. The type of cloud architecture is relevant as it determines the ability for data and resources to be shared.

Taking IoT and health records as an example: 1) health monitoring data from IoT devices may augment health records and 2) emergency detection based on multiple monitoring streams (heart-rate, pulse rate, temperature, and fall-detection) may need an emergency response. The monitored streams may be sent to care services such as ambulances and hospitals. Here, we may have health records hosted in private clouds while IoT-style health monitoring data and policy are hosted in public clouds. A healthcare practitioner will need access to both. However, this needs to be carefully regulated: ensuring the practitioner may only access the clinical records (private cloud) for patients they treat, and that the only monitoring data (public cloud) accessible to the practitioner is that which the patient has authorized, and may depend on the circumstances.

A research goal is to make public clouds sufficiently trustworthy, in order to meet the requirements of those, such as health services and government that deal with particularly sensitive data. Mechanisms providing strong data management assurances and controls (which we explore below) enable a wide range of new possibilities for applications and services.

In the meantime, however, the kind of scenario presented above motivates hybrid-clouds, where tenants manage the more sensitive aspects on their own (or dedicated) systems under their control, using a cloud-compatible service stack to integrate/interact with publicly accessible clouds. Clearly, the hybrid-cloud approach is rather blunt, as it entails physical infrastructure partitioning, and thus can preclude the nuanced sharing required by many application scenarios.

Consideration 5: In-Cloud Data Protection. This concerns the cloud provider protecting data within their service, by preventing data leakage: 1) during transmission; 2) during processing; and 3) when data is stored "in the cloud." In all cases, data should not flow to unauthorized parties, including cloud insiders as well as cloud users [95].

With respect to communication, some cloud providers now apply TLS internally within their infrastructure, including data centers to protect against any internal threats or security breaches.[5] This appears largely in response to recent highly publicized security breaches, such as those carried out by the U.S. National Security Agency.

The business model of cloud service provision is based on economies of scale, through services that share resources. For example, tenants may share the same physical machine by running above separate VMs during processing. Therefore, cloud providers ensure strong *isolation* between cloud tenants/users to prevent the leakage of data between them. This isolation can occur at different levels, including the OS (containers) [33], VM (hypervisor) [34], and in hardware (e.g., by leveraging Intel's proposed SGX CPU extensions [35]).

If storage is provided, depending on the level of isolation, the service offering might implicitly segregate all resources from others. Other levels of isolation may involve shared data storage infrastructure and software, such as shared databases, and thus rely on standard access control technologies (authentication and authorization)—see Consideration 2.

Cloud providers invest significant resources into ensuring strong access controls and complete isolation. Some are important for IoT-cloud, as well as for cloud service provision in general, but see Consideration 6. Concerns over the extent of provider access (by cloud insiders) do not only concern data that can objectively be considered highly sensitive. Rather, there may be laws that lead to particular data management obligations. Or, simply, there may be little trust in the cloud

[4]In practice, technical enabling sense—their terms of service—may disclaim all liability [29].

[5]For example, see [Online]. Available: http://wapo.st/1adFyAe, accessed on Apr. 2015.

provider, e.g., if they reside in a jurisdiction that lacks a robust data protection regime. In this case, the "thing" may decide to encrypt the data it uploads to the cloud, see Consideration 7.

Consideration 6: In-Cloud Data Sharing. We argued in Section II that the IoT vision entails *data sharing*, as required by applications and as controlled by their policy. Closed application "silos" should no longer be the norm and data should be able to flow as needed. For example, a heart-rate monitor and a motion detector may be separate "things" that upload their data to the cloud. In one usage, each "thing's" data stream is stored for the person being monitored, only accessible separately, and isolated from other "things" and other people's data. But policy-enforcing, management software for such medical applications may also be cloud-hosted and may need to input and process heart-rate, motion and other data to monitor patient wellbeing and detect and respond to emergencies, such as a collapse due to a heart attack.

In short, many benefits from the IoT vision are dependent on wide-ranging, open information sharing.

If each "thing's" data is uploaded to the cloud and isolated from other "things," as is the case in current cloud offerings (Consideration 5), the policy-enforcing agent described has no means of processing the data from multiple streams. To enable such a service, the system would be architected to suit some particular application; thus, favoring the very "silos" that preclude the wider vision.

We, therefore, have a requirement for both protection and sharing, according to policy, whereas cloud designs so far target strong protection without sharing.

There is ongoing research toward this. One approach being investigated is *information flow control* (IFC), where policy is defined to manage, specify, and control requirements for isolation and data sharing and to enforce them as data flows throughout a system [8]. IFC provides noninterference and nonleakage guarantees. Our own work has demonstrated IFC in a cloud context [36] to enforce data policy constraints within and between cloud applications and services; where flows are protected within an OS (at process level) [37] and across machines [38]. Enforcement is end-to-end, where the audit/provenance logs generated during IFC enforcement [37] can be used to demonstrate that policy has been complied with, whether user/application-specified, contractual, or regulatory.

IFC could help reassure people that even though their personal data has been uploaded to the cloud, it is protected and shared as they specify. This is an important concept, allowing users to retain control over their data, even when it has left their hands. Other relevant research is in the area of differential privacy and homomorphic encryption (see below) that aim to protect raw data while acknowledging the need for data sharing/processing.

The means by which tenants/users specify data sharing policy is also a concern. This may be in the form of standard templates, e.g., perhaps in line with service contracts, such as a contract between an individual and their healthcare provider, which can be adjusted to account for specific preferences [39].

Consideration 7: Encryption by "Things." "Things," users, and tenants could encrypt data before uploading to the cloud to: 1) prevent the provider having access to intelligible data; 2) prevent the provider being forced to disclose intelligible data to others, such as law enforcement agencies; 3) protect against the provider leaking data, due to misconfiguration, bugs, malicious insiders, etc.; 4) deal with differences in sensitivity for different data items; and/or 5) to protect data while in transit (specific data items, cf., the entire channel as per Consideration 1).

This approach results in the "things" having to manage all the security/data concerns, including key management which can be complex, particularly when many principals (in an IoT context, both users and "things") are involved [21]. For example, the data from a location sensor may be relevant to a number of applications. Assuming the sensor data can be encrypted before distribution, each time the set of authorized applications changes, all keys must be revoked, and new keys issued to all the relevant applications. This management burden hinders scalability. Furthermore, the issues concerning the resources required for encryption, as discussed in Consideration 1, are also relevant.

Cloud providers offer a range of services, typically relating to storage, analytics, and processing. Limiting a cloud provider's access to data reduces the range of services they can potentially offer. The wide-scale benefits of analytics over big data, cross-silo processing, etc., generally require access to intelligible data. Essentially, "thing"-encrypted data means that the provider can offer no more than a storage/IaaS service (or PaaS without any processing).

There is ongoing research into homomorphic encryption, which enables computation to be performed on encrypted data without access to plaintext [40], [41]; however, this is currently far from practicable. Therefore, for a provider to offer processing services, it must either have access to the data in intelligible form, or have access to decryption keys. In this case, encryption protects only against inadvertent leakage, and puts an onus on the provider to properly manage the keys.

In summary, "thing" managed encryption should be used with care since it may prevent the beneficial data composition and sharing described above in Consideration 6.

Consideration 8: Data Combination. While advocating both protection and beneficial sharing of "things" data, as in the examples above (Considerations 4 and 6), care should be taken over sharing. In IoT, "things" will act as data producers and consumers, generating or processing data of various levels of sensitivity. Some streams might be inherently sensitive, e.g., a location sensor on a personal device, or a person's heart-rate sensor. However, even if individual data streams are themselves benign, the application of data *in combination* can raise serious privacy and security concerns [42]. Such problems may be exacerbated by the use of cloud for IoT, as one of the motivations for cloud uptake is explicitly to enable data to be aggregated and used for a range of purposes, across the range of "things." Again, the motivation is to enable the wider, more imaginative IoT vision, by having more data for more accurate analysis, inferences, associations, personalization, and customization.

This concern relates to the tradeoff between the functional benefits of combining data, and the danger of revealing potentially sensitive information. From a privacy perspective: "Any information that distinguishes one person from another can be used for reidentifying anonymous data" [42]. There are technical approaches that can be used to limit the risks of data combination. For example, differential privacy techniques [43] aim at addressing the tradeoff by regulating the queries on a dataset to balance the provision of useful results with the probability of identifying individual records. Such techniques are beginning to be offered by cloud providers [44]. Furthermore, as homomorphic encryption techniques (Consideration 7) become more practicable, more value can be leveraged from data without access to the specifics. These techniques will contribute toward facilitating the wide-scale information sharing vision.

Although the cloud acting as data aggregator adds a risk of privacy violation through enabling richer datasets, and entails highly trusted providers, it also restricts data access to fewer places. This form of data "centralization," where data is accessed through the cloud yet may be distributed throughout the network, could enable aggregated, more focused *data management policy*, applicable across datasets, i.e., this centralization of data access means that such policy could apply more generally, accounting for data combination concerns, and be enforced through a common regime. This is in contrast to the fragmented approach where such policy might apply only within a particular IoT subsystem.

However, there is a more general problem in that it is difficult to anticipate all possible information leaks that might arise from combining data, and information sharing, in general. There is a clear need for some level of verifiable trust in the parties with which data is shared, including those hosting data.

Note that although we discuss this issue in an IoT-cloud context, the concerns extend far beyond, raising questions for society as a whole. Indeed, while there is much ongoing technical research into privacy in big data [45], the answers will not be purely technical, but also require properly aligned economic incentives, laws/regulation, and other social reforms [10].

V. Identity Management

The management of identity becomes an interesting problem for cloud-enabled IoT. In Section IV, we described how data is managed, a key aspect of which involves access controls, which tend to involve *authentication* and *authorization*, see Consideration 2. In the cloud-enabled IoT context, we identify two umbrella requirements with respect to identity management: from the provider and "thing" (tenant) perspectives.

Consideration 9: Identifying "Things." Identity management has been the subject of much work in terms of current enterprise services, i.e., there have been identity management schemes, often single sign-on [46], [47], across cloud service, and application providers such as Microsoft Services, Google Services, Facebook, etc. For example, consider identity federation technology such as Microsoft Passport and CardSpace, Information Card, OpenID, Liberty Alliance, and Higgins [48].

However, all of these concern cloud services as they are today. Users interact with the tenant's application, and the tenant is hosted by a cloud provider. Issues of identity concern who interacts with the applications and cloud resources.

The IoT brings additional considerations, as it involves more than the well-defined tenant–software-provider relationship. In IoT the provider could potentially receive the data of a number of "things," that belong to and/or produce data on a tenant/end user, i.e., an individual could have several hundred data sources uploading to the provider. Some of these might go through applications dedicated to them (in a similar manner to today—which may simplify the problem); others might be uploading to shared applications or directly to the cloud platform. It follows that there must be a mechanism for providers to determine to which tenants and/or end-users the data streams belong.

The first step is to be able to identify the "things," which might be a new subsystem comprising a large number of nodes [49]. There is work in the area, e.g., having an architecture that groups "things" to enable the common application of policy [50]. This is akin to an IoT subsystem, but where the group exists purely for identity/policy management purposes. TPM [23], as mentioned in Consideration 2 may also assist.

After authentication, it must be possible to both specify and identify to which tenant/user the "thing" belongs, i.e., there must be suitably flexible, scalable identity mechanisms that tie the "thing" to the relevant tenant/user account. Authorization and other management policy are built on this.

A consideration is that some "things" could: 1) be shared and/or 2) generate data that is relevant to a number of different tenants. For example, home monitoring and control (domotic) systems have user-specific policies, requiring people to be identified. A proximity sensor in a house could identify when different members of the family are near to it—there needs to be some way of determining the context (e.g., relating to which family member) in which the sensor is operating. Each person might have different preferences and uses for the data generated by the device. It may also be necessary to temporarily account for "strangers," such as visiting tradesman.

Issues are further complicated by actuators: knowing which "things" to actuate, and when, to effect some change in the physical environment. It becomes particularly important that the right actions are triggered for the right person. Also, conflicts might arise, since physical changes can affect different people, who might have different preferences. In the home example, different members of the family may have different temperature preferences, and thus policies over thermostat control could conflict. In the case of simultaneous policies applying and conflicting, detection and resolution mechanisms are needed.

In the cloud, there is an intrinsic tension between the end-users' requirement for privacy and the application providers' economic interests (as the sayings goes: "if you are not paying for it, you're not the customer; you're the product being sold").[6] While data is valuable, so too is identity since it can ground various attributes and inferences, leading to targeted advertising, changes in health premiums, and so forth [51]. These are general, identity-based concerns, based on identities that exist in the real world (e.g., identifying an individual, group of people (family), or business). However, even the identity of "things" can release sensitive information. For instance, the fact that someone owns a particular device could imply they have some medical condition [26].

[6]This quote is generally attributed to Andrew Lewis. [Online]. Available: https://twitter.com/andlewis/status/24380177712, accessed on Apr. 2015.

From a human rights/legal point of view, it has been argued [52], [53] that IoT information should be considered as part of an individual's identity and protected in the same manner as their physical identity. Cameron [54] defined seven fundamental laws for digital identities: 1) user control and consent; 2) minimal disclosure for constrained use; 3) justifiable parties; 4) directed identity; 5) interoperability; 6) human integration; and 7) consistent experience across context.

Consideration 10: Identifying the Provider. The inverse consideration is that "things" must interact with the correct cloud service. Making sure the correct "thing" (or the relevant gateway component) sends the information to the right cloud service *a priori* is typically a configuration issue, where fixed/common configuration mechanisms are appropriate for some situations; e.g., for the range of "things" owned by the same individual or business.

However, there are nuances. For example, if a "thing" generates data relevant to multiple applications (hosted on different cloud providers), how should the "thing" know which data to send where, and when? The "thing" would need the capability (credentials) to effect the relevant cloud interactions, and maintain policy determining with which cloud services to interact. Alternatively, this could be managed by the cloud service, coordinating and distributing data across "things," applications, and clouds, but this requires shared resources that can account for, and resolve policies of multiple actors.

Furthermore, there will also be occasions when these concerns will require runtime negotiation, e.g., when an individual first interacts with a sensor. How should this be managed? How do they transfer their policies, and dictate where that data should flow? These are complex issues all of which need consideration.

VI. Managing Scale for the IoT-Cloud

Cloud services exist to exploit economies of scale. A key offering of the cloud is *elasticity*, where resources can be rapidly scaled up or down in response to changes in demand. This functionality is highly attractive to tenants, as it allows for cost-effective improvements in application/service availability.

Consideration 11: Increase in Load. Traditionally, the elasticity of cloud services was aimed at resourcing web applications, where an "end-user" represents a thread or instance of a web-application. In the IoT space, there is a vast increase not only in the number of clients (i.e., "things") that the cloud must interact with, but also in terms of data volume, velocity, and variety [55]. Cloud services must, therefore, be able to manage a range and scale of devices that potentially produce data far in excess of today's volume and peak loads. The failure to scale leads to availability issues, which can have serious implications by limiting access to data or preventing the cloud from coordinating and mediating the "things."

Scale represents a real challenge. We currently see that cloud-enabled applications are often unable to rely on elasticity alone to deal with periods of extreme demand even in a web context, e.g., many clients attempting to book popular event tickets at the moment of release [56]. In such situations, other techniques (e.g., queuing systems and/or customized architectures) need to be employed to manage such loads. For IoT, issues of managing at such scale could well be the norm.

It is also important to account for any performance overhead brought by the security mechanisms.

Consideration 12: Logging at Large Scale. Logs are important for ensuring that systems are functioning as expected, and for demonstrating compliance with regulations, laws, and contracts (see Section IX and [95]).

Since many more "things" may be interacting with cloud services, logging and audit suffer from problems of scale. This is from a number of perspectives: in terms of what the cloud provider must record; the fact that logs might be decentralized among the "things"; that different systems/verticals will vary in what is (and needs to be) recorded; and what can sensibly be interpreted from log data, which may be large, federated, and potentially in different formats. In such a context, it makes some sense to push the log data from "things" to the cloud, to provide a better overview of state, but this will necessarily incur cost, in terms of processing, storage, and transmission.

It becomes important to be able to define policy that captures the audit goals or the legal requirements through the different layers of the cloud stack, while minimizing the amount of data captured to acquire the relevant information [57]. However, most of the logs available in the cloud are an aggregation of the logs of various cloud components, coming from webservers, the OS, databases, etc. These logs are system-centric. In terms of the wider IoT vision, tenants, and users will also require logs pertaining to their data, not just system status. Thus, logging mechanisms must evolve to capture information in a more data-centric fashion [58].

Another consideration is managing the location of log information across the range of "things." One approach is to centralize log information, e.g., [59] proposes an approach to reliably collect logs from various sources, removing duplicate/unnecessary information, while accounting for failure or disconnections. Such an approach seems highly suited to cloud services. The alternative is to develop analysis tools that can work over decentralized log data [60]. This shows promise as it accounts for the coordination and *ad hoc* aspects of IoT. Perhaps a hybrid approach is sensible.

Another consideration is managing the location of log information across the range of "things." One approach is to centralize log information, e.g., [59] proposes an approach to reliably collect logs from various sources, removing duplicate/unnecessary information, while accounting for failure or disconnections. Such an approach seems highly suited to cloud services. The alternative is to develop analysis tools that can work over decentralized log data [60]. This shows promise as it accounts for the coordination and *ad hoc* aspects of IoT. Perhaps a hybrid approach is sensible.

There is also the tension between the volume of log information and the associated storage and processing overheads. Toward this, work includes dynamically modifying the verbosity of the log when there is a potential threat [61], or *a posteriori* editing of the log to remove unnecessary information [62].

In general, log analysis in large complex distributed systems still presents many unsolved challenges [61]. Certainly, more work is needed in addressing such issues in the context of an IoT-cloud.

VII. Malicious Things

The previous sections broadly consider aspects of management. As mentioned (see Section IV), cloud providers already protect their infrastructure from a range of different attacks, through having appropriate access controls, isolation, encryption and sanitization functionality (for PaaS/SaaS), etc. To reiterate, cloud providers have clear incentives to maintain a secure infrastructure, because: 1) their business model depends on sharing infrastructure and 2) failure to provide adequate security measures will result in negative publicity and thus a loss in reputation and business.

Given our focus on IoT-cloud, we do not explore the protection measures that apply to cloud-computing services, in general; [63] and [64] provide overviews. Note also that any security mechanisms developed to address the IoT-cloud security considerations we raise, may be subjected to attack. Such attacks would be solution-specific, thus any analysis would only be relevant within the context of the specific approach and implementation. Therefore, in this section, we focus on two situations, specific to cloud-enabled IoT, where the attacks come from malicious (or compromised) "things." This is a real concern; for instance, a wide variety of smart home appliances have been discovered to be the source of large-scale spam attacks.[7]

Consideration 13: Malicious "Things"—Protection of Provider. The cloud provider will maintain various access, and other controls, to protect against specific attacks, e.g., a rogue "thing" attempting to exploit the service, perhaps through some sort of injection attack. Even if attacks are successful, cloud isolation mechanisms offer containment, limiting their fallout. Such attacks are not unlike the security concerns of the cloud as it is today.

Previous sections have explored how IoT dramatically increases scale, where there is the potential for a vast number of "things" to interact directly with a provider. Thus, one clear IoT-cloud vulnerability is cloud *denial of service* (DoS), which could potentially be launched from a large number of compromised "things." Cloud services are naturally *elastic*, designed to rapidly scale up/down resources in response to increases in demand, but still remain vulnerable to DoS [56]. Therefore, there is a need to explore more advanced DoS techniques in light of the fact that IoT greatly increases the scope of such an attack, particularly as "things" become more integrated with cloud services.

Consideration 14: Malicious "Things"—Protection of Others. Since the cloud can operate as a mediator and coordinator between "things," it offers potential in terms of improving security across the IoT ecosystem. This is because the cloud provides a natural "choke-point" between "things," in which security policy can be implemented and enforced.[8]

Requiring input data to pass through a validation process allows the cloud to effectively disconnect (or ignore inputs from) "things" that are detected as compromised. This also helps ensure data integrity, as only valid data (in terms of rate/format)—rather than that from a faulty, compromised (rogue), or inappropriate (but perhaps nonmalicious) "thing"—can enter a (possibly shared) database or flow to others via the cloud. Furthermore, there is scope for the cloud to be used more proactively, e.g., by issuing control messages to "things" to turn them off (or adjust some parameters) where necessary, or perhaps to trigger software/firmware updates.

Warnings could also be issued to alert those that own, use, or rely on those "things" that are determined to be faulty or compromised. These could be high-level (human-readable) or low-level (in an M2M context), as appropriate.

A fundamental consideration is in determining the "things" that have been compromised. This will be relevant at different levels, depending on the circumstances; for instance, approaches could involve determining the malicious or untrustworthy nodes in a network [65], analyzing the data outputs, patterns of behavior or reputation of a "thing" [66], or perhaps involve human intervention, e.g., reporting a device as stolen. Work is required on developing such techniques, in line with new developments in technologies and their uses.

VIII. Trust in the Cloud Provider: Certification

Any prospective tenant, before committing to the use of cloud services, needs to consider the trustworthiness of the provider. This has many dimensions, as discussed throughout this paper. Here, we focus on aspects of certification—what should be certified and how? Section IX extends this discussion to the demonstration of compliance with regulations and laws.

Consideration 15: Certification of Cloud Service Providers. Certification can be about system configuration, and the associated management processes (particularly management of risk), both at a human level (e.g., engineer involvement, regulating physical access) and a more technical level (e.g., whether security standards are adhered to). A number of regulated sectors, such as in government and health, may only use cloud service offerings that are certified as being compliant across the relevant regulatory landscape, U.K.'s G-Cloud,[9] and in the U.S. FedRAMP[10] and HIPAA,[11] being representative. Even those operating in less regulated sectors will have an interest in their provider demonstrating compliance with various standards, such as ISO/IEC 27001:2013 [67] on information security, to provide a degree of assurance.

Currently, certification is often the only available way to demonstrate compliance with regulations [68]. The automating of certification processes has been considered [69]–[71], but certification is currently a human-centered process that assesses system behavior at the time of the audit. Any changes to a deployment can trigger the need for recertification, which is often a timely and costly process. Furthermore, the advent or installation of new technology or architecture needs to go

[7][Online]. Available: http://investors.proofpoint.com/releasedetail.cfm?releaseid=819799, accessed on Apr. 2015.

[8]Of course, things may also interact directly, without the use of cloud services. But architectures and services could leverage cloud-based protection capabilities when/where appropriate to limit the scope of attacks, even if the cloud does not mediate every interaction between "things." See also Section X.

[9][Online]. Available: https://www.gov.uk/government/publications/g-cloud-security-accreditation-application, accessed on Apr. 2015.

[10][Online]. Available: http://cloud.cio.gov/fedramp, accessed on Apr. 2015.

[11][Online]. Available: http://www.hhs.gov/ocr/privacy, accessed on Apr. 2015.

through a certification process, thus introducing similar inefficiencies. Overall such constraints do not align to the general, flexible vision of the cloud, let alone IoT-cloud.

It may be possible to formalize some aspects of compliance, e.g., with regard to some aspects of security [72], [73]; however, such work explicitly recognizes the difficulties given the lack of cloud provider transparency. Another issue concerns service composition, in the sense that even if two systems are individually secure, the composition of the services may not be [73]. This is particularly relevant in a cloud context, where cloud services may be composed (see Consideration 18). Furthermore, if the cloud operates as a coordinator of "things," then the provider may bear extra responsibility to ensure such co-ordinations are appropriate, e.g., in terms of data combination (see Consideration 8).

Therefore, more technical means of defining the appropriate cloud-provider behavior and demonstrating compliance is needed, as discussed in the rest of this section and in Section IX.

Consideration 16: Trustworthiness of Cloud Services. A general concern is how much trust can be placed in a cloud service provider; i.e., that they will properly 1) secure their service; 2) ensure it is correctly configured; 3) report leakages/issues; and 4) use data only for their intended purposes. Key to building trust is providing some degree of visibility/transparency over the cloud service. Section IX discusses how this might be enhanced through audit, including when using external, third-party cloud services and controlling where data is located in order to abide by regulations.

Recent developments in hardware technologies [74] enable new levels of trust, providing TPM [23] and remote attestation for cloud computing [75]. These can work to increase the level of trust that tenants have in the provider; for instance, by enabling data integrity and confidentially to be guaranteed regardless of the platform on which the data is processed [76], or to provide guarantees concerning the physical location of data [77] (see Consideration 19). Such techniques are reaching maturity, e.g., IBM is rolling out a scalable TPM-based cloud platform [75], [78].

It is likely the case that end-users are more willing to trust well-established and known cloud providers, rather than those with little history or reputation, such as startups offering cloud-hosted applications. Several projects have focused on preventing the misuse or leakage of data by cloud applications through complex isolation mechanisms [79], or by incorporating IFC [37], [80] (see Consideration 6), which enables the control policy to be attached to data (potentially by "things," tenants, or providers) in order to control the flow of data, and to generate audit logs. More generally, having mechanisms that limit data mismanagement are crucial to enabling the wide-scale vision of information sharing underpinning the IoT.

IX. Provider Transparency: Compliance

Some data management constraints arise from the nature and functionality of the applications and services. Others are a result of regulation (e.g., data protection legislation) and contractual obligation [e.g., service-level agreements (SLAs)]. Rather than the cloud-provider being a "black-box," in both situations it is advantageous to have some visibility into a provider's operations, be it for compliance purposes, or more generally, to give some surety that data is properly managed.

For all the considerations in this section, the concerns will become particularly pertinent for the IoT-cloud, given it entails a vast increase in data producers, consumers, and service providers; where data and services may be used/reused for a number of purposes.

Consideration 17: Demonstrating Compliance Using Audit. Cloud service providers issue contracts (SLAs) indicating the terms and conditions of cloud tenants' usage. There is currently often little or no provision for negotiation of the service conditions [29], nor any automated means of demonstrating compliance with all the terms within a contract. More generally, tenants may have obligations with respect to data management; e.g., data protection regulations in the EU apply to data considered personal [29].

Trustworthy audit services are relevant for cloud tenants, end-users, and providers. Tenants and users can be assured that the platform is performing as it should be (and that they are getting what they pay for), and for providers such services help detect data leaks, misconfigurations, and other security issues. Audit is also relevant for verifying compliance with law/regulation [81]. Clearly, such information helps reinforce accountability [82], be it to show some fault of the provider, or conversely to absolve their responsibility, when a leak has been claimed falsely. Furthermore, such data would also be useful more generally, e.g., by public-sector bodies charged with advising on and enforcing information-related policy (such as the U.K. Information Commissioner's Office).

The recent surge in cloud uptake and the evolving IoT market has meant there is beginning to be some work on audit. For example, Massonet *et al.* [83] propose a framework whereby a cloud provider generates an audit log so that the cloud tenant is able to demonstrate his compliance with location-related regulation, and in an IoT context, the Infineon TPM[12] uses hardware-based cryptography to produce tamper-proof audit logs. It is important that audit mechanisms are developed, not only to handle the scale of the IoT vision (Section VI) but also to ensure that all relevant aspects are captured, and that access to audit information is properly regulated (log data can be sensitive). All of these pose challenges, given the way IoT services are composed, where data (and services) can be used/reused for different purposes.

Consideration 18: Responsibility for Composite Services. It is common for cloud service providers to leverage a number of third-party services. Other cloud platforms could be involved in service provision, e.g., building a PaaS offering over IaaS provider, as is the case for Heroku PaaS that runs over Amazon IaaS, providing the feature set for tenants to build SaaS applications.[13] Other third-party services may also be involved such as those providing log archiving and analytic tools. It follows that the legal obligations between tenants, end-users, providers, and the providers' entire supply chain can be unclear [84]. Policies related to data location (see Consideration 19) may be

[12][Online]. Available: https://www.infineon.com/cms/en/applications/chip-card-security/internet-of-things-security/audit-and-accountability, accessed on Apr. 2015.

[13][Online]. Available: https://www.heroku.com/customers, accessed on Apr. 2015 for a list of commercial entities already running over such services.

relevant, in addition to the more general concern of who has access to data.

Some recent work is addressing these concerns, e.g., Henze *et al.* [85] propose an annotation, audit, and negotiation system for multiparty layered cloud offering (SaaS, PaaS, and IaaS) to meet tenant specified requirements. However, such issues become even more complex in an IoT context, where services will be composed more dynamically.

As an initial step forward, more transparency and visibility as to the specifics of how cloud services are composed and provisioned would assist in determining the appropriate responsibility and regulation frameworks, to which technical composition mechanisms can aspire.

A further consideration is *application level* composition, i.e., where "things," including those cloud based, are brought together by application/user-level concerns; particularly, where it is the *composition itself* that brings about a vulnerability. In this context, issues concerning policy authoring, validation, and conflict resolution are relevant; for details, see [86], consideration 2, and [27] for a practical illustration involving a lighting system, IFTT,[14] and Facebook. The possibility for dynamic, perhaps unforeseen compositions raises interesting risk and obligation management challenges.

Consideration 19: Compliance With Data Location Regulations. The broad IoT vision is for "things" to interact, wherever they are, when and where appropriate. There are, however, real concerns relating to the physical (geo)location of data.

This issue is less apparent when considering "things" in isolation, as "things" tend naturally to be grounded in some physical environment, space (e.g., sensor networks in a building or city) or coupled with an individual (e.g., a mobile phone). Cloud services, however, deliberately aim to be *globally centralized* [87], generally accessible from anywhere and everywhere. Thus, in mediating between "things," the nature of cloud provisioning means that data could potentially be moved and stored, and "things" orchestrated and controlled, across geographic boundaries. There are practical concerns, most obviously in terms of law, when data (or control) flow span national borders [95].

As a result, we have seen much political rhetoric calling for regional clouds (such as a Europe-only cloud), particularly post-Snowden, in an attempt to circumvent various governmental agencies and for competitive advantage—see [84] for a full analysis of the related societal issues. Practically, laws and best practices that constrain data flows based on geography are an attempt to give certainty and visibility as to the legal regime and management principles that apply to data. We have explored the technical considerations of constraining data by location for legal purposes in [10], [88], and [95].

Hybrid-clouds are marketed as a solution, where data with location-based constraints remains on the tenant's self-managed infrastructure. While this addresses issues of location, this can be costly and limits the wider benefits of the cloud. Furthermore, they hinder the flexible sharing underpinning the wider vision of IoT (Consideration 4).

[14] [Online]. Available: https://ifttt.com, accessed on Apr. 2015.

It is apparent that more control mechanisms are needed to address the fact that "things" are local, but the cloud-based data services and analytics are potentially global. We raise this issue here as it represents a real, practical hurdle that must be overcome in order to realize wider IoT vision.

X. Decentralized Clouds: A Future Trend

Our discussion so far has concerned the cloud of today, where the cloud in effect represents—from the tenant perspective—a global, but centralized infrastructure [87]. This is our focus as it represents the current state of the art, which is already beginning to be used to support IoT and big data applications.

Moving forward, however, there is ongoing research into decentralized cloud computing. In general terms, this involves pushing the cloud services toward the edges of the network, toward and closer to the "things." Key motivations of such research are to reduce the latency, delay, jitter, network congestion, and resource usage that naturally arise from local/mobile "things" interacting with the global centralized cloud. Work on decentralization is not considered a replacement for the global cloud—which will still have a place as aggregator, coordinator, and a pool of resources—but rather represents the means to better deal with the challenges associated with the local ("thing")–global interplay.

There are differences among the proposed approaches. Fog computing [87], [89] describes more of a distributed computation approach, akin to edge [90] or grid computing, where certain service functionality is composed from among "things" and cloud services, at various levels, data flowing where appropriate (e.g., pre/post computation). Cloudlets [91] are concerned with mobile cloud computing, where personal VMs (e.g., stored on mobile devices) can be offloaded onto more fixed infrastructure in the environment, e.g., that situated in a cafe or shopping mall, in order to leverage general cloud resources, when possible. This is to bring various efficiencies over the device acting either by itself or in conjunction with the more distant global cloud. Droplets [92] enable similar capabilities, but focus specifically on small, well-defined, highly customized VMs (*unikernels* [93]), which can enable personal- or even application/service-specific clouds. As well as efficiency, an explicit design goal of droplets is to enable a user to be in control of their personal data and services: an individual could precisely define the functionality and content of each droplet, and decide when/where and by whom each droplet is hosted.

Consideration 20: Impact of Cloud Decentralization on Security. The concept of the decentralized cloud raises interesting security considerations. It could reduce the attack surface of the global cloud, and perhaps the vulnerability to DoS, because fewer "things" would directly interact with remote cloud services.

Conversely, the smaller, decentralized entities are likely to be less robust, e.g., in terms of the security mechanisms that can be applied, and more vulnerable to DoS, due to the lack of resource elasticity. Furthermore, decentralization paves the way for more targeted attacks, e.g., directed toward an individual, cf., the global cloud provider; and the data flows moving in/out of the more controlled, global cloud infrastructure will occur

TABLE I

CONSIDERATIONS, SECURITY FOCUS (C=CONFIDENTIALITY, I=INTEGRITY, A=AVAILABILITY) AND CURRENT STATUS GREEN =SOME MATURITY IN APPROACHES; AMBER =SOME RESEARCH EXISTS, MORE WORK NEEDED; RED =RELATIVELY UNEXPLORED AREA

#	Consideration	Focus	Status
1	Secure communications	C, I	Work is advanced and existing techniques can be leveraged. IoT could benefit from lighter weight schemes, particularly where cryptography is involved.
2	Access controls for IoT-Cloud	C	Standard mechanisms can be used. IoT adds complexity due to the scale and dynamism of "thing" access.
3	Identifying sensitive data	C	Largely a nontechnicalconcern, but has an impact on how policies are defined.
4	Public, private or hybrid?	C, A	Currently blunt partitioning is supported, but emerging research will allow for more flexible deployments that facilitate data sharing.
5	In-cloud data protection	C	There are strong isolation techniques available and providers employ general access controls. More flexible approaches are needed for interapplication sharing to be possible (see 6, below).
6	In-cloud data sharing	C, A	Interapplication sharing is needed for IoT but currently is not part of the cloud philosophy.
7	Encryption by "things"	C, I	Encryption techniques are mature, but this approach precludes most computations on protected data and involves complex key management. Ongoing work into homomorphic encryption will assist. Work on lightweight encryption mechanisms are being developed and will therefore require robust testing and analysis.
8	Data combination	C	Some techniques exist to prevent user reidentification, but much more work is needed.
9	Identifying "things"	C	Existing work on identity management can be leveraged for IoT, but more experience at a larger scale is needed to determine suitability and/or limitations.
10	Identifying the provider	C	The basic issues are mostly architectural or configuration concerns. Some outstanding issues remain when resources are shared or where decisions need to be made at runtime.
11	Increase in interactions and data load	A	Cloud services manage elasticity well, but resource expansion is not unlimited. Peak IoT loads are unknown, but possibly controlled by economics (payment/ownership).
12	Logging at large scale	C, I, A	Currently logging is low-level and system-centered. More work is needed on logging and processing tools for applications and users.
13	Malicious "things"—protection of provider	C, I, A	Existing techniques can be deployed.
14	Malicious "things"—protection of others	C, I, A	There are potentially techniques that can assist. Experience is needed of cloud services operating across IoT subsystems.
15	Certification of cloud service providers	C, I, A	This is currently manual and static, leading to delays when updates are required. Research is needed on automatic certification processes, possibly including hardware-based solutions.
16	Trustworthiness of cloud services	C, I, A	An emerging field with ongoing research. Experience of practical implementation is needed.
17	Demonstrating compliance using audit	C, I, A	Currently, the compliance of cloud providers to their contractual obligations is not demonstrated convincingly. Research is needed, and IoT will add additional complexity.
18	Responsibility for composite services	C, I, A	The legal implications of the use of third-party and other services are unresolved. Such usage is not as yet transparent to tenants and/clients. More work is needed concerning user and application-level policy aspects.
19	Compliance with data location regulations	C, I, A	Currently not enforceable except at coarse granularity. There is research in IFC that can assist, but the concepts are not yet commercially deployed.
20	Impact of cloud decentralization	C, I, A	This is an emerging field, where the current focus is on functionality. More attention is needed regarding security.

more frequently, thus raising additional management concerns. Coordinating security mechanisms, such as software updates and security patches, and identity management present real challenges in highly federated environments. Depending on how decentralized clouds come to be realized, "things" possibly may become more embedded within the cloud service, which could increase the severity of an attack.

More generally, as the systems environment becomes decentralized, it may be the case that more "things" *directly* interact, rather than rely on cloud services—particularly as "things" become more powerful. Such interactions require the means for flexible management. There is work on infrastructure toward this, such as SBUS [94]: a decentralized, peer-to-peer-based communications infrastructure that aims at policy-driven interactions. Such functionality appears useful in managing all combinations of "things" interacting with other "things," "things" with clouds, and clouds with clouds.

XI. SUMMARY

Concern over data security in cloud computing is already seen as inhibiting the adoption of public cloud services for a number of sectors and organizations [31]. Legal and regulatory issues are also emerging [95], such as the location of data and identifying the jurisdictions under which they fall [84].

With this background, we have considered the use of cloud technology for IoT, to reduce the propensity for application "silos" and enable the beneficial sharing of data. Cloud services can clearly hold and process the data of "things," and components that manage "things" and combine data streams from "things" are highly amenable to being hosted within the cloud. Cloud and IoT potentially present vast scope for considering security. In this paper, we have identified and described 20 security-related considerations within the following broad range of concerns:

1) issues of data transport to/from cloud services and data management in the cloud (Sections III and IV);
2) issues associated with identity management (Section V);
3) issues associated with the scale of IoT (Section VI);
4) issues arising from malicious "things" (Section VII);
5) issues of certification, trust, and compliance with regulations and contractual obligations (Sections VIII and IX);
6) issues arising from further decentralization into multiple clouds, fog services, etc. (Section X).

Table I lists these 20 considerations, indicating where current, standard existing technologies can be used (green), where more work is required but the problems are reasonably well understood (amber), and where significant research is needed to understand and solve the problems (red).

We see data sharing as an intrinsic part of the IoT philosophy, yielding many benefits. Of course, sharing must be controlled according to policy, which must be informed by the possible consequences of unconsidered data sharing. Cloud services have been designed with *protection* (isolation) as the dominant concern, with far less consideration given to *sharing*. A promising approach to providing both data protection and sharing is to augment principal-centered access control technologies with those that focus on the properties of the data. IFC, for instance, can prevent data leakage while relaxing the strong isolation that currently prevents data sharing between applications [8], [37].

Only if controlled data sharing can be supported by public cloud services can the wider IoT-vision be realized.

References

[1] M. Weiser, "Ubiquitous computing," *Computer*, vol. 26, no. 10, pp. 71–72, 1993.

[2] P. Middleton, P. Kjeldsen, and J. Tully, *Forecast: The Internet of Things, Worldwide*. Gartner, 2013 [Online]. Available: http://www.gartner.com/document/2625419?ref=QuickSearch&sthkw=G00259115.

[3] S. Jankowski, J. Covello, H. Bellini, J. Ritchie, and D. Costa, "The Internet of Things: Making Sense of the Next Mega-Trend," Goldman Sachs, New York, Tech. Rep., 2014 [Online]. Available: http://www.goldmansachs.com/our-thinking/outlook/internet-of-things/iot-report.pdf

[4] A. Zanella, "Internet of Things for smart cities," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 22–32, Feb. 2014.

[5] J. Bacon *et al.*, "Personal and social communication services for health and lifestyle monitoring," in *Proc. 1st Int. Conf. Global Health Challenges (Global Health'12) IARIA DataSys 2012*, Venice, Italy, 2012, pp. 41–48.

[6] M. Kovatsch, S. Mayer, and B. Ostermaier, "Moving application logic from the firmware to the cloud: Towards the thin server architecture for the Internet of Things," in *Proc. 6th Int. Conf. Innov. Mobile Internet Serv. Ubiq. Comput. (IMIS)*, 2012, pp. 751–756.

[7] E. P. Leverett, "Quantitatively assessing and visualising industrial system attack surfaces," M.S. thesis, Dept. Comput. Lab., Univ. Cambridge, Cambridge, U.K., 2011.

[8] J. Bacon, D. Eyers, T. Pasquier, J. Singh, I. Papagiannis, and P. Pietzuch, "Information flow control for secure cloud computing," *IEEE Trans. Netw. Serv. Manage.*, vol. 11, no. 1, pp. 76–89, Mar. 2014.

[9] P. Ohm, "Broken promises of privacy: Responding to the surprising failure of anonymization," *UCLA Law Rev.*, vol. 57, no. 6, pp. 1701–1777, Aug. 2010.

[10] J. Singh *et al.*, "Regional clouds: Technical considerations," Univ. Cambridge, Cambridge, U.K., Tech. Rep. UCAM-CL-TR-863, 2014 [Online]. Available: http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-863.pdf, accessed on Jul. 21, 2015.

[11] "Overview of the Internet of Things," ITU Telecommun. Standard. Sector, Tech. Rep. Y.2060, Jun. 2012.

[12] J. Mineraud, O. Mazhelis, X. Su, and S. Tarkoma, "A gap analysis of Internet-of-Things platforms," 2015, arXiv:1502.01181 [Online]. Available: http://arxiv.org/abs/1502.01181

[13] S. Abdelwahab, B. Hamdaoui, M. Guizani, and A. Rayes, "Enabling smart cloud services through remote sensing: An Internet of Everything enabler," *IEEE Internet Things J.*, vol. 1, no. 3, pp. 276–288, 2014.

[14] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, 2010.

[15] D. Kozlov, J. Veijalainen, and Y. Ali, "Security and privacy threats in IoT architectures," in *Proc. 7th Int. Conf. Body Area Netw. (BodyNets)*, 2012, pp. 256–262.

[16] W. A. Jansen, "Cloud hooks: Security and privacy issues in cloud computing," in *Proc. 44th Hawaii Int. Conf. Syst. Sci. (HICSS)*, 2011, pp. 1–10.

[17] T. Dierks and C. Allen, "The TLS protocol version 1.0," IETF, Tech. Rep., RFC 5246, 1999 [Online]. Available: https://tools.ietf.org/html/rfc5246

[18] D. McGrew and E. Rescorla, "Datagram transport layer security (DTLS) extension to establish keys for the secure real-time transport protocol (SRTP)," Internet Engineering Task Force (IETF), RFC 5764, 2010 [Online]. Available: http://datatracker.ietf.org/doc/rfc5764/

[19] S. L. Keoh, S. Kumar, and H. Tschofenig, "Securing the Internet of Things: A standardization perspective," *IEEE Internet Things J.*, vol. 1, no. 3, pp. 265–275, 2014.

[20] P. Urien, "LLCPS: A new security framework based on TLS for NFC P2P applications in the Internet of Things," in *Proc. IEEE Consum. Commun. Netw. Conf. (CCNC)*, 2013, pp. 845–846.

[21] R. J. Anderson, *Security Engineering: A Guide to Building Dependable Distributed Systems*, 2nd ed. Hoboken, NJ, USA: Wiley, 2008.

[22] Z. Durumeric *et al.*, "The matter of heartbleed," in *Proc. Internet Meas. Conf. (IMC)*, 2014, pp. 475–488.

[23] T. Morris, "Trusted platform module," in *Encyclopedia of Cryptography and Security*. New York, NY, USA: Springer, 2011, pp. 1332–1335.

[24] C. Lesjak, T. Ruprechter, J. Haid, H. Bock, and E. Brenner, "A secure hardware module and system concept for local and remote industrial embedded system identification," in *Proc. Emerg. Technol. Factory Autom. (ETFA)*, 2014, pp. 1–7.

[25] M. Hutter and R. Toegl, "A trusted platform module for near field communication," in *Proc. Int. Conf. Syst. Netw. Commun. (ICSNC)*, 2010, pp. 136–141.

[26] J. Singh and J. Bacon, "On middleware for emerging health services," *J. Internet Serv. Appl.*, vol. 5, no. 6, pp. 1–34, 2014.

[27] N. Dhanjani. (2013). Hacking lightbulbs: Security evaluation of the Philips hue personal wireless lighting system [Online]. Available: http://www.dhanjani.com/docs/Hacking%20Lighbulbs% 20Hue %20Dhanjani%202013.pdf, accessed on Jul. 21, 2015.

[28] R. M. Savola and H. Abie, "Metrics-driven security objective decomposition for an e-health application with adaptive security management," in *Proc. Int. Workshop Adapt. Secur.*, 2013, p. 6.

[29] C. J. Millard, Ed., *Cloud Computing Law*. London, U.K.: Oxford Univ. Press, 2013.

[30] N. Dhanjani. (2013). Reconsidering the perimeter security argument [Online]. Available: http://www.dhanjani.com/docs/Reconsidering%20the%20Perimeter%20Security%20Argument.pdf, accessed on Jul. 21, 2015.

[31] Intel IT Centre, *What's Holding Back the Cloud?*, Intel Corporation, 2012 [Online]. Available: http://www.intel.co.uk/content/dam/www/public/us/en/documents/reports/whats-holding-back-the-cloud-peer-research-report2.pdf

[32] T. Pasquier, B. Shand, and J. Bacon, "Information flow control for a medical web portal," in *Proc. IADIS Int. Conf. e-Society*, IADIS, Lisbon, Portugal, 2013, pp. 123–130 [Online]. Available: http://www.iadisportal.org/e-society-2013-proceedings

[33] S. Soltesz, H. Pötzl, M. E. Fiuczynski, A. Bavier, and L. Peterson, "Container-based operating system virtualization: A scalable, high-performance alternative to hypervisors," in *ACM SIGOPS Operating Syst. Rev.*. New York, NY, USA: ACM, 2007, vol. 41, no. 3, pp. 275–287.

[34] P. Barham *et al.*, "Xen and the art of virtualization," in *SIGOPS Operating Syst. Rev.*, 2003, vol. 37, no. 5, pp. 164–177.

[35] I. Anati, S. Gueron, S. P. Johnson, and V. R. Scarlata, "Innovative technology for CPU based attestation and sealing," Intel Corporation, 2013 [Online]. Available: https://software.intel.com/sites/default/files/article/413939/hasp-2013-innovative-technology-for-attestation-and-sealing.pdf.

[36] T. F. J.-M. Pasquier, J. Singh, and J. Bacon, "Information flow control for strong protection with flexible sharing in PaaS," in *Proc. Int. Workshop Future of PaaS (IC2E)*, 2015, pp. 279–282.

[37] T. Pasquier, J. Singh, D. Eyers, and J. Bacon, CamFlow: Managed data-sharing for cloud services, 2015, arXiv:1506.04391 [Online]. Available: http://arxiv.org/abs/1506.04391

[38] J. Singh, T. Pasquier, J. Bacon, and D. Eyers, "Integrating middleware with information flow control," in *Proc. Int. Conf. Cloud Eng. (IC2E)*, 2015, pp. 54–59.

[39] J. Singh and J. Bacon, "Governance in patient-centric healthcare," in *Proc. Int. Conf. Inf. Soc. (i-Society)*, 2010, pp. 502–509.

[40] M. Naehrig, K. Lauter, and V. Vaikuntanathan, "Can homomorphic encryption be practical?" in *Proc. 3rd ACM Workshop Cloud Comput. Secur. (CCSW)*, 2011, pp. 113–124.

[41] D. Hrestak and S. Picek, "Homomorphic encryption in the cloud," in *Proc. 37th Int. Conv. Inf. Commun. Technol. Electron. Microelectron. (MIPRO)*, 2014, pp. 1400–1404.

[42] A. Narayanan and V. Shmatikov, "Myths and fallacies of personally identifiable information," *Comm. ACM*, vol. 53, pp. 24–26, 2010.

[43] C. Dwork, "Differential privacy: A survey of results," in *Theory and Applications of Models of Computation*. New York, NY, USA: Springer, 2008, pp. 1–19.

[44] Ú. Erlingsson, V. Pihur, and A. Korolova, "RAPPOR: Randomized aggregatable privacy-preserving ordinal response," in *Proc. SIGSAC Conf. Comput. Commun. Secur.*, 2014, pp. 1054–1067.

[45] M. Jensen, "Challenges of privacy protection in big data analytics," in *Proc. IEEE Int. Congr. Big Data*, 2013, pp. 235–238.

[46] D. Recordon and D. Reed, "OpenID 2.0: A platform for user-centric identity management," in *Proc. 2nd ACM Workshop Digital Identity Manage.*, 2006, pp. 11–16.

[47] R. Morgan, S. Cantor, S. Carmody, W. Hoehn, and K. Klingenstein, "Federated security: The Shibboleth approach," *Educause Q.*, vol. 27, pp. 12–17, 2004.

[48] T. El Maliki and J.-M. Seigneur, "A survey of user-centric identity management technologies," in *Proc. Int. Conf. Emerg. Secur. Inf. Syst. Technol. (SecureWare)*, 2007, pp. 12–17.

[49] P. N. Mahalle, B. Anggorojati, N. R. Prasad, and R. Prasad, "Identity authentication and capability based access control (IACAC) for the Internet of Things," *J. Cyber Secur. Mobility*, vol. 1, no. 4, pp. 309–348, 2013.

[50] P. De Leusse, P. Periorellis, T. Dimitrakos, and S. K. Nair, "Self managed security cell, a security model for the Internet of Things and services," in *Proc. 1st Int. Conf. Adv. Future Internet*, 2009, pp. 47–52.

[51] Boston Consulting Group, *The Value of our Digital Identity*. Liberty Global Inc., 2012 [Online]. Available: http://www.libertyglobal.com/PDF/public-policy/The-Value-of-Our-Digital-Identity.pdf

[52] P. De Hert, "A right to identity to face the Internet of Things?," UNESCO 2008 [Online]. Available: http://portal.unesco.org/ci/en/files/25857/12021328273de_Hert-Paul.pdf/de%2BHert-Paul.pdf

[53] P. De Hert, S. Gutwirth, A. Moscibroda, D. Wright, and G. G. Fuster, "Legal safeguards for privacy and data protection in ambient intelligence," *Pers. Ubiq. Comput.*, vol. 13, no. 6, pp. 435–444, 2009.

[54] K. Cameron, *The Laws of Identity*. Microsoft Corporation, 2005 [Online]. Available: https://msdn.microsoft.com/en-us/library/ms996456.aspx

[55] L. Douglas, "3D data management: Controlling data volume, velocity and variety," Gartner Inc. [Online]. Available: http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity-and-Variety.pdf

[56] S. Yu, "DDoS attack and defence in cloud," in *Distributed Denial of Service Attack and Defense*. New York, NY, USA: Springer, 2014, pp. 77–93.

[57] R. K. Ko *et al.*, "TrustCloud: A framework for accountability and trust in cloud computing," in *Proc. IEEE World Congr. Serv. (SERVICES)*, 2011, pp. 584–588.

[58] R. K. Ko, M. Kirchberg, and B. S. Lee, "From system-centric to data-centric logging-accountability, trust & security in cloud computing," in *Proc. Defense Sci. Res. Conf. Expo. (DSR)*, 2011, pp. 1–4.

[59] A. Rabkin and R. H. Katz, "Chukwa: A system for reliable large-scale log collection," in *Proc. 24th Int. Conf. Large Install. Syst. Admin. (LISA)*, USENIX, 2010, vol. 10, pp. 1–15.

[60] A. Rabkin, M. Arye, S. Sen, V. Pai, and M. J. Freedman, "Making every bit count in wide-area analytics," in *HotOS*, May 2013, 6 pp.

[61] A. Oliner, A. Ganapathi, and W. Xu, "Advances and challenges in log analysis," *Comm. ACM*, vol. 55, no. 2, pp. 55–61, 2012.

[62] A. Elyasov, I. Prasetya, and J. Hage, "Log-based reduction by rewriting," Utrecht Univ., Tech. Rep. UU-CS-2012-013, 2012.

[63] S. Subashini and V. Kavitha, "A survey on security issues in service delivery models of cloud computing," *J. Netw. Comput. Appl.*, vol. 34, no. 1, pp. 1–11, 2011.

[64] M. Jensen, J. Schwenk, N. Gruschka, and L. Iacono, "On technical security issues in cloud computing," in *Proc. IEEE Int. Conf. Cloud Comput. (CLOUD'09)*, 2009, pp. 109–116.

[65] T. Moore, J. Clulow, S. Nagaraja, and R. Anderson, "New strategies for revocation in ad-hoc networks," in *Proc. 4th Eur. Conf. Secur. Privacy Ad-Hoc Sens. Netw.*, 2007, pp. 232–246.

[66] P. Michiardi and R. Molva, "Core: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks," in *Proc. IFIP TC6/TC11 6th Joint Work. Conf. Commun. Multimedia Secur. Adv. Commun. Multimedia Secur*. Norwell, MA, USA: Kluwer, 2002, pp. 107–121.

[67] "ISO/IEC JTC 1/SC 27 IT-security technology—Security techniques—Information security management systems-requirements," ISO, Tech. Rep. 27001, 2013.

[68] A. Sunyaev and S. Schneider, "Cloud services certification," *Commun. ACM*, vol. 56, no. 2, pp. 33–36, 2013.

[69] R. Accorsi, D.-I. L. Lowis, and Y. Sato, "Automated certification for compliant cloud-based business processes," in *Business & Information Systems Engineering*. New York, NY, USA: Springer, 2011, vol. 3, no. 3, pp. 145–154.

[70] A. Muñoz and A. Mana, "Bridging the GAP between software certification and trusted computing for securing cloud computing," in *Proc. 9th World Congr. Serv. (SERVICES)*, 2013, pp. 103–110.

[71] T. Kunz, A. Selzer, and U. Waldmann, "Automatic data protection certificates for cloud-services based on secure logging," in *Trusted Cloud Computing*. New York, NY, USA: Springer, 2014, pp. 59–75.

[72] S. A. de Chaves, C. B. Westphall, and F. R. Lamin, "SLA perspective in security management for cloud computing," in *Proc. 6th Int. Conf. Netw. Serv. (ICNS)*, 2010, pp. 212–217.

[73] K. Bernsmed, M. G. Jaatun, P. H. Meland, and A. Undheim, "Security SLAs for federated cloud services," in *Proc. 6th Int. Conf. Availability Reliab. Secur. (ARES)*, 2011, pp. 202–209.

[74] Intel, Software guard extensions programming reference, Tech. Rep. 329298-001US, 2013 [Online]. Available: https://software.intel.com/sites/default/files/329298-001.pdf, accessed on Jul. 21, 2015.

[75] S. Berger, R. Cáceres, K. A. Goldman, R. Perez, R. Sailer, and L. van Doorn, "vTPM: Virtualizing the trusted platform module," in *Proc. Secur. Symp.*, 2006, pp. 305–320.

[76] A. Baumann, M. Peinado, and G. Hunt, "Shielding applications from an untrusted cloud with haven," in *Proc. 11th USENIX Conf. Oper. Syst. Des. Implement. (OSDI)*, 2014, pp. 267–283.

[77] K. R. Jayaram, D. Safford, U. Sharma, V. Naik, D. Pendarakis, and S. Tao, "Big ideas paper: Trustworthy geographically fenced hybrid clouds," in *Proc. ACM/IFIP/USENIX Middleware*, 2014, pp. 37–48.

[78] S. Berger, K. Goldman, D. Pendarakis, D. Safford, E. Valdez, and M. Zohar, "Scalable attestation: A step toward secure and trusted clouds," in *Proc. Int. Conf. Cloud Eng. (IC2E)*, 2015, pp. 185–194.

[79] S. Lee, E. L. Wong, D. Goel, M. Dahlin, and V. Shmatikov, "πBox: A platform for privacy-preserving apps," in *Proc. 10th USENIX Symp. Netw. Syst. Des. Implement.*, 2013, pp. 501–514.

[80] K. Singh, S. Bhola, and W. Lee, "xBook: Redesigning privacy control in social networking platforms," in *Proc. USENIX Secur. Symp.*, 2009, pp. 249–266.

[81] K. Ruan, J. Carthy, T. Kechadi, and M. Crosbie, "Cloud forensics," in *Advances in Digital Forensics VII*. New York, NY, USA: Springer, 2011, pp. 35–46.

[82] A. Haeberlen, "A case for the accountable cloud," *SIGOPS Oper. Syst. Rev.*, vol. 44, no. 2, pp. 52–57, 2010.

[83] P. Massonet, S. Naqvi, C. Ponsard, J. Latanicki, B. Rochwerger, and M. Villari, "A monitoring and audit logging architecture for data location compliance in federated cloud infrastructures," in *Proc. Int. Symp. Parallel Distrib. Process. Workshops PhD Forum (IPDPSW)*, 2011, pp. 1510–1517.

[84] K. Hon, C. Millard, C. Reed, J. Singh, I. Walden, and J. Crowcroft, Policy, legal and regulatory implications of a Europe-only cloud, School of Law, Queen Mary Univ. London, London, U.K., Tech. Rep. 191, 2015 [Online]. Available: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2527951, accessed on Jul. 21, 2015.

[85] M. Henze, R. Hummen, and K. Wehrle, "The cloud needs cross-layer data handling annotations," in *Proc. Secur. Privacy Workshops (SPW)*, 2013, pp. 18–22.

[86] J. Singh, "Controlling the dissemination and disclosure of healthcare events," Ph.D. dissertation, Computer Lab., Univ. Cambridge, London, U.K.: Tech. Rep. TR 770, 2009.

[87] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the Internet of Things," in *Proc. 1st MCC Workshop Mobile Cloud Comput.*, 2012, pp. 13–16.

[88] T. Pasquier and J. Powles, "Expressing and enforcing location requirements in the cloud using information flow control," in *Proc. IC2E Int. Workshop Legal Tech. Issues Cloud Comput. (Claw'15)*, 2015, pp. 410–415.

[89] K. Hong, D. Lillethun, U. Ramachandran, B. Ottenwälder, and B. Koldehofe, "Mobile fog: A programming model for large-scale applications on the Internet of Things," in *Proc. 2nd SIGCOMM Workshop Mobile Cloud Comput. (MCC)*, 2013, pp. 15–20.

[90] A. Davis, J. Parikh, and W. E. Weihl, "Edgecomputing: Extending enterprise applications to the edge of the internet," in *Proc. 13th Int. World Wide Web Conf. Altern. Track Papers*, 2004, pp. 180–187.

[91] M. Satyanarayanan, P. Bahl, R. Caceres, and N. Davies, "The case for VM-based cloudlets in mobile computing," *IEEE Pervasive Comput.*, vol. 8, no. 4, pp. 14–23, 2009.

[92] J. Crowcroft, A. Madhavapeddy, M. Schwarzkopf, T. Hong, and R. Mortier, "Unclouded vision," in *Distributed Computing and Networking*. New York, NY, USA: Springer, 2011, pp. 29–40.

[93] A. Madhavapeddy and D. J. Scott, "Unikernels: The rise of the virtual library operating system," *Commun. ACM*, vol. 57, no. 1, pp. 61–69, 2014.

[94] J. Singh, D. Eyers, and J. Bacon, "Policy enforcement within emerging distributed, event-based systems," in *Proc. ACM Distrib. Event Based Syst. (DEBS'14)*, 2014, pp. 246–255.

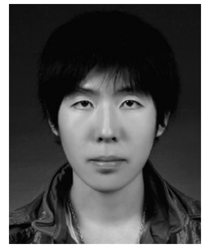
[95] J. Singh, J . Powles, T. Pasquier, and J. Bacon, "Data flow management and compliance in cloud computing," *IEEE Cloud Comput. Magaz. SI Legal Clouds*, 2015, to be published.

Jatinder Singh is currently a Senior Research Associate with the Computer Laboratory, University of Cambridge, Cambridge, U.K. His research interests include concern management and control in distributed systems, particularly regarding cloud and the Internet of Things.

Thomas Pasquier received the M.Phil degree from Cambridge University, Cambridge, U.K., and is currently working toward the Ph.D. degree at the University of Cambridge, Cambridge, U.K. His M.Phil. degree included a project on "prevention of identity inference in deidentified medical records."

He is currently a Research Assistant with the University of Cambridge.

Jean Bacon is currently a Professor of distributed systems with the University of Cambridge, Cambridge, U.K., where she leads the Opera Research Group, with a focus on open, large-scale, secure, and widely distributed systems.



Hajoon Ko is currently working toward the Ph.D. degree at the University of Cambridge, Cambridge, U.K.

David Eyers is currently a Senior Lecturer with the University of Otago, Dunedin, New Zealand, and a Visiting Research Fellow with the Cambridge Computer Laboratory, University of Cambridge, Cambridge, U.K.